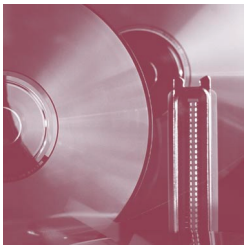
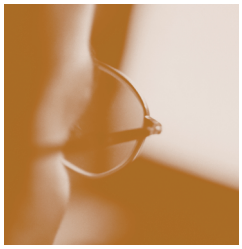
ASG-Bridge™ User's Guide

Version: 6.0

Publication Number: BCX0200-60

Publication Date: February 2002





ASG Worldwide Headquarters Naples, Florida USA | asg.com
1333 Third Avenue South, Naples, Florida 34102 USA Tel: 941.435.2200 Fax: 941.263.3692 Toll Free: 1.800.932.5536

ASG Documentation/Product Enhancement Fax Form

Please FAX comments regarding ASG products and/or documentation to (941) 263-3692.

Company Name	Telephone Number	Site ID	Contact name

Product Name/Publication	Version #	Publication Date
Product:		
Publication:		
Tape VOLSER:		

Enhancement Request:

ASG Support Numbers

ASG provides support throughout the world to resolve questions or problems regarding installation, operation, or use of our products. We provide all levels of support during normal business hours and emergency support during non-business hours. To expedite response time, please follow these procedures.

Please have this information ready:

- Product name, version number, and release number
- List of any fixes currently applied
- Any alphanumeric error codes or messages written precisely or displayed
- A description of the specific steps that immediately preceded the problem
- The severity code (ASG Support uses an escalated severity system to prioritize service to our clients. The severity codes and their meanings are listed below.)
- Verify whether you received an ASG Service Pack for this product. It may include information to help you resolve questions regarding installation of this ASG product. The Service Pack instructions are in a text file on the distribution media included with the Service Pack.

If You Receive a Voice Mail Message:

1. Follow the instructions to report a production-down or critical problem.
2. Leave a detailed message including your name and phone number. A Support representative will be paged and will return your call as soon as possible.
3. Please have the information described above ready for when you are contacted by the Support representative.

Severity Codes and Expected Support Response Times

Severity	Meaning	Expected Support Response Time
1	Production down, critical situation	Within 30 minutes
2	Major component of product disabled	Within 2 hours
3	Problem with the product, but customer has work-around solution	Within 4 hours
4	"How-to" questions and enhancement requests	Within 4 hours

ASG provides software products that run in a number of third-party vendor environments. Support for all non-ASG products is the responsibility of the respective vendor. In the event a vendor discontinues support for a hardware and/or software product, ASG cannot be held responsible for problems arising from the use of that unsupported version.

Business Hours Support

Your Location	Phone	Fax	E-mail
United States and Canada	800.354.3578	941.263.2883	support@asg.com
Australia	61.2.9460.0411	61.2.9460.0280	support.au@asg.com
England	44.1727.736305	44.1727.812018	support.uk@asg.com
France	33.141.028590	33.141.028589	support.fr@asg.com
Germany	49.89.45716.222	49.89.45716.400	support.de@asg.com
Singapore	65.332.2922	65.337.7228	support.sg@asg.com
All other countries:	1.941.435.2200		support@asg.com

Non-Business Hours - Emergency Support

Your Location	Phone	Your Location	Phone
United States and Canada	800.354.3578		
Asia	65.332.2922	**Japan/Telecom**	0041.800.9932.5536
Australia	0011.800.9932.5536	**Netherlands**	00.800.3354.3578
Denmark	00.800.9932.5536	**New Zealand**	00.800.9932.5536
France	00.800.3354.3578	**Singapore**	001.800.3354.3578
Germany	00.800.3354.3578	**South Korea**	001.800.9932.5536
Hong Kong	001.800.9932.5536	**Sweden/Telia**	009.800.9932.5536
Ireland	00.800.9932.5536	**Switzerland**	00.800.9932.5536
Israel/Bezeq	014.800.9932.5536	**Thailand**	001.800.9932.5536
Japan/IDC	0061.800.9932.5536	**United Kingdom**	00.800.3354.3578
		All other countries	1.941.435.2200

ASG Web Site

Visit http://www.asg.com, ASG's World Wide Web site.

Submit all product and documentation suggestions to ASG's product management team at http://www.asg.com/asp/emailproductsuggestions.asp.

If you do not have access to the web, FAX your suggestions to product management at (941) 263-3692. Please include your name, company, work phone, e-mail ID, and the name of the ASG product you are using. For documentation suggestions include the publication number located on the publication's front cover.

Contents

Preface

This *ASG-Bridge User's Guide* provides instructions for using ASG-Bridge (herein called Bridge). Bridge simplifies software system conversions such as those required by date conversions and euro currency compliance. It provides solutions for batch, CICS, and IMS applications written in COBOL, PL/I, and Assembler that use VSAM, sequential files, IMS and, IDMS databases.

Allen Systems Group, Inc. (ASG) provides professional support to resolve any questions or concerns regarding the installation or use of any ASG product. Telephone technical support is available around the world, 24 hours a day, 7 days a week.

ASG welcomes your comments, as a preferred or prospective customer, on this publication or on any ASG product.

About this Publication

This publication consists of these chapters:

- Chapter 1, "Overview," provides an overview of bridging features and benefits.
- Chapter 2, "Concepts," describes bridging concepts.
- Chapter 3, "Getting Started," describes how to set up your Bridge environment.
- Chapter 4, "Creating Bridge Parameters and Bridge Rules," describes how to define parameters and Bridge Rules.
- Chapter 5, "Building a Bridge Definition," describes how to identify each file to Bridge.
- Chapter 6, "Generating Executable Bridge Routines," describes how to submit Bridge Definitions to generate executable Bridge Routines.
- Chapter 7, "Cross-Reference Feature," describes how you can view a summary of all Bridge Definitions in a specified AKR along with their respective attributes.
- Chapter 8, "Importing/Exporting Parameters, Rules, and Definitions," describes how to import and export Bridge Parameters, Rules, and Definitions.
- Chapter 9, "Bridge Event Log,"describes how to use the Bridge Event Log.
- Chapter 10, "Testing Your Bridge Routine," describes how to test Bridge Routines against your data. This testing does not require access to your application programs.
- Chapter 11, "Program Integration," describes how to integrate Bridge into your program.
- Chapter 12, "Record-Level API," describes how to use the Record-level API feature when Bridge's standard run-time facility does not support the application or environment that needs bridging.

Related Publications

The documentation library for ASG-Bridge consists of these publications (where *nn* represents the product version number):

- *ASG-Bridge Installation Guide* (BCX0300-*nn*) provides information on installation and maintenance of ASG-Bridge.
- *ASG-Bridge User's Guide* (BCX0200-*nn*) contains information regarding this product, which enables field expansion for program source code, without being required to simultaneously expand the fields in files or databases.
- *ASG-Center Installation Guide* (CNX0300-*nn*) contains installation and maintenance information for ASG-Center, the common set of libraries shared by the ASG-Existing Systems Workbench suite of products.

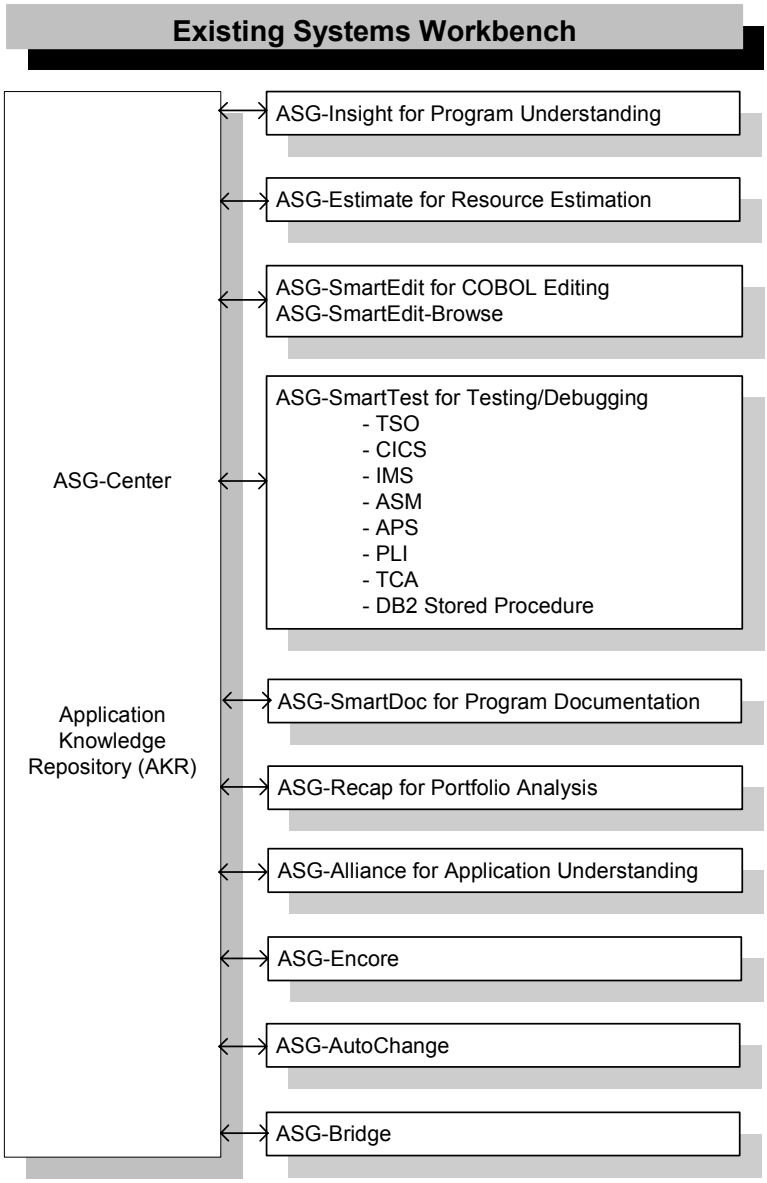
Note:

To obtain a specific version of a publication, contact the ASG Service Desk.

ASG-Existing Systems Workbench (ASG-ESW)

ASG-ESW (herein called ESW) is an integrated suite of components designed to assist organizations in enhancing, redeveloping, or re-engineering their existing systems. ESW products use the Application Knowledge Repository (AKR) to store source program analysis information generated by the Analytical Engine. Figure 1 represents the components of ESW.

Figure 1 • ASG Existing Systems Workbench



This table contains the name and description of each ESW component:

ESW Product	Herein Called	Description
ASG-Alliance	Alliance	The application understanding component that is used by IT professionals to conduct an analysis of every application in their environment. Alliance supports the analysis and assessment of the impact of change requests upon an entire application. Alliance allows the programmer/ analyst to accurately perform application analysis tasks in a fraction of the time it would take to perform these tasks without an automated analysis tool. The impact analysis from Alliance provides application management with additional information for use in determining the resources required for application changes.
ASG-AutoChange	AutoChange	The COBOL code change tool that makes conversion teams more productive by enabling quick and safe changes to be made to large quantities of code. AutoChange is an interactive tool that guides the user through the process of making source code changes.
ASG-Bridge	Bridge	The bridging product that enables field expansion for program source code, without being required to simultaneously expand the fields in files or databases. Because programs are converted in smaller groups, or on a one-by-one basis, and do not require file conversion, testing during the conversion process is simpler and more thorough.
ASG-Center	Center	The common platform for all ESW products. Center provides the common Analytical Engine to analyze the source program and store this information in the AKR. This common platform provides a homogeneous environment for all ESW products to work synergistically.

ESW Product	Herein Called	Description
ASG-Encore	Encore	The program re-engineering component for COBOL programs. Encore includes analysis facilities and allows you to extract code based on the most frequently used re-engineering criteria. The code generation facilities allow you to use the results of the extract to generate a standalone program, a callable module, a complement module, and a CICS server. Prior to code generation, you can view and modify the extracted Logic Segment using the COBOL editor.
ASG-Estimate	Estimate	The resource estimation tool that enables the user to define the scope, determine the impact, and estimate the cost of code conversion for COBOL, Assembler, and PL/I programs. Estimate locates selected data items across an application and determines how they are used (moves, arithmetic operations, and compares). Time and cost factors are applied to these counts, generating cost and personnel resource estimates.
ASG-Insight	Insight	The program understanding component for COBOL programs. Insight allows programmers to expose program structure, identify data flow, find program anomalies, and trace logic paths. It also has automated procedures to assist in debugging program abends, changing a computation, and resolving incorrect program output values.
ASG-Recap	Recap	The portfolio analysis component that evaluates COBOL applications. Recap reports provide function point analysis and metrics information, program quality assessments, intra-application and inter-application comparisons and summaries, and historical reporting of function point and metrics information. The portfolio analysis information can also be viewed interactively or exported to a database, spreadsheet, or graphics package.
ASG-SmartDoc	SmartDoc	The program documentation component for COBOL programs. SmartDoc reports contain control and data flow information, an annotated source listing, structure charts, program summary reports, exception reports for program anomalies, and software metrics.

ESW Product	Herein Called	Description
ASG-SmartEdit	SmartEdit	The COBOL editing component that can be activated automatically when the ISPF/PDF Editor is invoked. SmartEdit provides comprehensive searching, inline copybook display, and syntax checking. SmartEdit allows you to include an additional preprocessor (for example, the APS generator) during syntax checking. SmartEdit supports all versions of IBM COBOL, CICS, SQL, and CA-IDMS.
ASG-SmartTest	SmartTest	The testing/debugging component for COBOL, PL/I, Assembler, and APS programs in the TSO, MVS Batch, CICS (including file services), and IMS environments. SmartTest features include program analysis commands, execution control, intelligent breakpoints, test coverage, pseudo code with COBOL source update, batch connect, disassembled object code support, and full screen memory display.

Invoking ESW Products

The method you use to invoke an ESW product depends on your system setup. If you need assistance to activate a product, see your systems administrator. If your site starts a product directly, use the ISPF selection or CLIST as indicated by your systems administrator. If your site uses the ESW screen to start a product, initiate the ESW screen using the ISPF selection or CLIST as indicated by your systems administrator and then typing in the product command on the command line.

The product names can also vary depending on whether you access a product directly or through ESW. See "ESW Product Integration" on page xvi for more information about using ESW.

To initialize ESW products from the main ESW screen, select the appropriate option on the action bar pull-downs or type the product shortcut on the command line.

Product Name	Shortcut	ESW Pull-down Options
Alliance	AL	Understand ▸ Application
AutoChange	CC	Change ▸ Conversion Set
Bridge	BR	Change ▸ ASG-Bridge
Encore (Re-engineer)	EN	Re-engineer ▸ Program
Estimate	ES	Measure ▸ ASG-Estimate
Insight (Understand)	IN	Understand ▸ Program
Recap (Portfolio Analysis)	RC	Measure ▸ Portfolio
SmartDoc (Document)	DC	Document ▸ Program
SmartEdit	SE	Change ▸ Program **Or** Change ▸ Program with Options
SmartTest	ST	Test ▸ Module/Transaction

ESW Product Integration

Because ESW is an integrated suite of products, you are able to access individual ESW products directly or through the main ESW screen. As a result, you might see different fields, values, action bar options, and pull-down options on a screen or pop-up depending on how you accessed the screen or pop-up.

Certain ESW products also contain functionality that interfaces with other ESW products. Using SmartTest as an example, if Alliance is installed, SmartTest provides a dynamic link to Alliance that can be used to display program analysis information. If Insight is installed and specified during the analyze, the Insight program analysis functions are automatically available for viewing logic/data relationships and execution path. For example, the Scratchpad option is available on the Options pull-down if you have Insight installed. Access to these integrated products requires only that they be installed and executed in the same libraries.

Examples

Example 1. Figure 2 shows the Encore Primary screen that displays when you access Encore directly.

The Encore Primary screen contains these eight action bar menu items: File, View, Extract, Generate, Search, List, Options, and Help.

Figure 2 • Encore Primary Screen

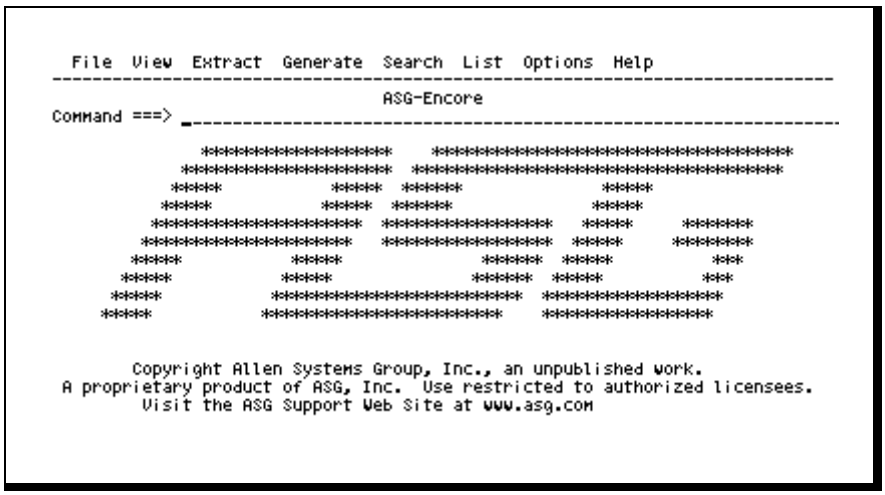
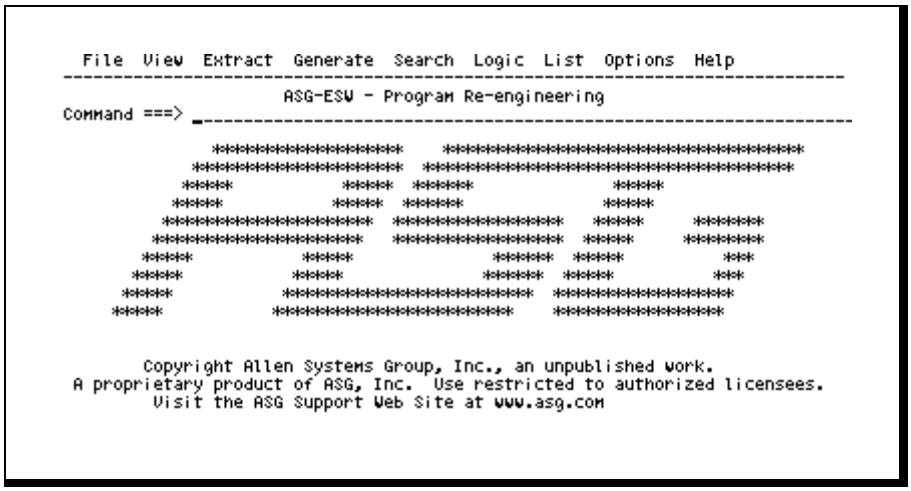


Figure 3 shows the Encore Primary screen that displays when you access Encore through ESW by selecting Re-engineer ▸ Program from the ESW action bar menu. Notice that the Primary screen name changes to ASG-ESW - Program Re-engineering when you enter Encore through ESW. Also, the Logic menu item displays if Insight is installed.

Figure 3 • ESW Encore Primary Screen



Example 2. Figure 4 shows the File - Analyze Submit pop-up that displays when you access SmartTest directly. Figure 5 shows the File - Analyze Submit pop-up that displays when you access SmartTest through ESW.

Notice that the Analyze features field in Figure 5 lists additional ESW products than shown on Figure 4. This field is automatically customized to contain the ESW products you have installed on your system.

The actions shown on these screens also vary. For example, the D action (ASG-SmartDoc Options) is available on the File - Analyze Submit screen if the SmartDoc product is installed on your system. In Figure 4, the ASG-SmartDoc Options action is not available.

Figure 4 • File - Analyze Submit Screen

```
                          File - Analyze Submit
Command ===> ____________________________________________________________

          E - Edit JCL                                S - Submit JCL

Compile and link JCL (PDS or sequential):
   Data set name  'USER12.REL.CNTL(VIAPCOBC)'

Analyze features (Y/N):
   ASG-SmartTest:  Y   Extended Analysis:  N

AKR data set name 'USER12.GENERAL.AKR'
AKR program name                       (if overriding PROGRAM-ID)

Analyze options:
   ------------------------------------------------------------------------
   ------------------------------------------------------------------------
   ------------------------------------------------------------------------

Compile? (Y/N) . . . . . . . . . . Y     (Y if needed by features)
Link load module reusable? (Y/N) Y
```

Figure 5 • File - Analyze Submit Screen (Accessed through ESW)

```
                          File - Analyze Submit
Command ===> ____________________________________________________________

          E - Edit JCL     S - Submit JCL     D - ASG-SmartDoc Options

Compile and link JCL (PDS or sequential):
   Data set name  'USER12.REL.CNTL(HTEST)'

Analyze features (Y/N):
   ASG-Insight:  Y   ASG-SmartTest:  Y   Extended Analysis:  N
   ASG-SmartDoc:  N   ASG-Encore:  N
AKR data set name 'USER12.GENERAL.AKR'
AKR program name  __________           (if overriding PROGRAM-ID)

Analyze options:
   ------------------------------------------------------------------------
   ------------------------------------------------------------------------
   ------------------------------------------------------------------------

Compile? (Y/N) . . . . . . . . . . Y     (Y if needed by features)
Link load module reusable? (Y/N) Y     (ASG-SmartTest
```

Publication Conventions

ASG uses these conventions in technical publications:

Convention	Represents
ALL CAPITALS	Directory, path, file, dataset, member, database, program, command, and parameter names.
Initial Capitals on Each Word	Window, field, field group, check box, button, panel (or screen), option names, and names of keys. A plus sign (+) is inserted for key combinations (e.g., Alt+Tab).
lowercase italic monospace	Information that you provide according to your particular situation. For example, you would replace *filename* with the actual name of the file.
Monospace	Characters you must type exactly as they are shown. Code, JCL, file listings, or command/statement syntax. Also used for denoting brief examples in a paragraph.
Vertical Separator Bar (|) with underline	Options available with the default value underlined (e.g., Y|N).

1 Overview

This chapter presents an overview of Bridge and contains these sections:

What is Bridging?

Bridging is a compliance conversion strategy that is program-centric. It permits you to perform on-the-fly conversions of your programs either individually or within groups without requiring file conversions. In bridging, converted programs process unconverted files by converting the file records as they are read and written. The file remains in an unconverted state.

What is Bridge?

Bridge is a dynamic bridging tool you can use to build non-intrusive Bridge Routines that interface between converted programs and non-converted files. Bridge automatically invokes these Bridge Routines to convert records read and written from non-converted files during program execution, deferring the need to convert files. Bridge consists of an online, interactive administration facility, and a dynamically invoked automatic bridging component.

Benefits of Bridge

The benefits of using Bridge for your date or euro currency conversion projects include:

- Simplified conversion planning. Planning becomes program-centric rather than data-centric.
- Minimized complexity and risk of conversions.
- Deferred deadline for converting data files indefinitely.
- Reduced coordination problems with using EDI (electronic data interchange) and other outside data.
- Allows compliant and non-compliant programs to share the same data before and after conversion.
- Guaranteed seamless access to non-compliant historical and backup data.
- Eliminates secondary maintenance requirements after the file is converted.

Implementing Bridge

To implement Bridge, you define each file structure and its respective conversion fields. Bridge uses these definitions to generate executable bridging routines that you invoke to convert records read and written by the program. These executable Bridge Routines are external to the application programs.

Bridge automatically intercepts all program I/O operations and converts the data to and from the new format as needed. The program gets a converted view of the non-converted file.

These are the two major phases of implementing Bridge technology:

- Generating the Bridge Routines from user-defined Bridge Definitions.
- Integrating the Bridge Routines with a converted program.

Generating Bridge Routines

Bridge provides an online, interactive ISPF-based facility that allows you to define and generate executable Bridge Routines.

To define an executable Bridge Routine

1. Create a Bridge Definition to define a file that requires bridging.
2. Identify conversion fields that require bridging.

3 For each conversion field, select the from/to bridging rule.

4 If the selected Bridge Rule requires parameters, review or define their values.

5 If the file contains multiple record formats, designate Test Fields and assign Field Test names and values that can be used to identify each record format. Create Record Definitions that consist of one or more Field Tests.

6 Associate the Bridge Definition with a Generate library (existing load library) where generated executable Bridge Routines will be stored.

7 Generate the executable Bridge Routine from the Bridge Definition.

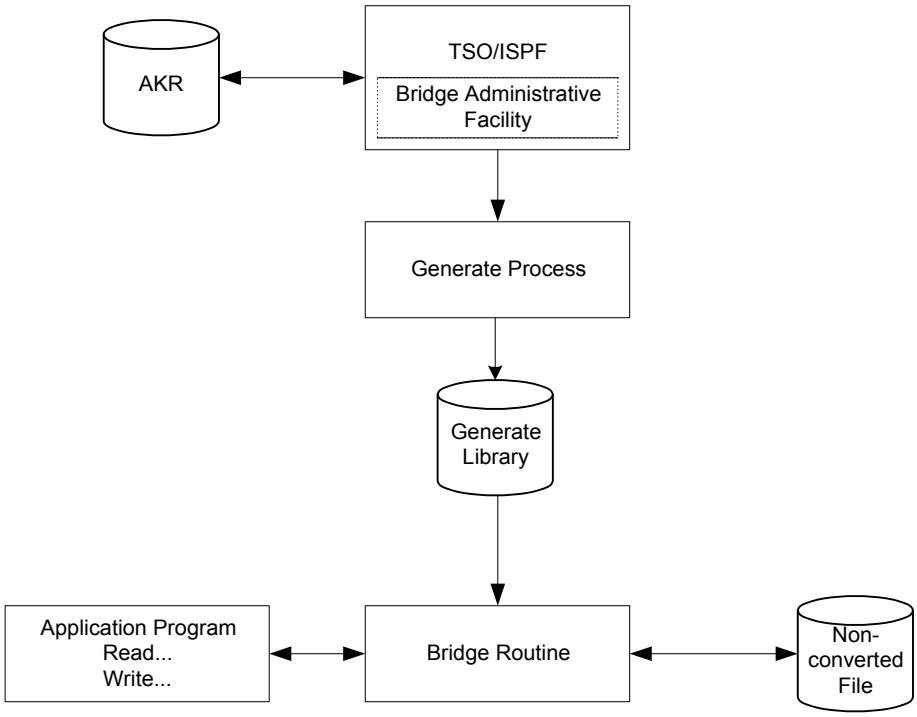
Integrating Bridge Routines with a Converted Program

To trigger Bridge, you add a single call to the initializing routine of a converted program. This call identifies the files that you want bridged and associates them with their Bridge Routines.

After the data file and all of the programs that access the file are converted you can tell Bridge that the file conversion is completed. This deactivates bridging for that file.

Figure 6 illustrates how Bridge works.

Figure 6 • How Bridge Works



2 Concepts

This chapter describes Bridge concepts and contains these sections:

Several strategies exist that address date and euro currency conversion processes. Of these, Bridge employs the dynamic bridging strategy. Dynamic bridging combines field expansion and windowing techniques, allowing you to expand field definitions within programs without requiring the simultaneous expansion of related files and databases. This strategy, as implemented by Bridge, is the simplest, safest, and most cost effective strategy available today.

Dynamic bridging strategy involves modifying program logic to accommodate expanded or converted fields. This strategy also incorporates interpretive logic within the program to check whether the program requires the bridging technique. The bridging routine determines whether input or output records contain compliant or non-compliant fields immediately after a datastore read or before a datastore write. The I/O Bridge logic then converts the fields appropriately, based on the current status of each specific datastore being accessed.

Bridge simplifies this strategy even further by providing the required bridging logic external to the program code. You use a single CALL statement to invoke Bridge, vastly minimizing your programming time.

Benefits of Bridging Strategy

The bridging strategy for file conversion projects that works best with large applications is the dynamic bridging strategy. This is because it allows you to upgrade a large number of programs over time while you prepare for the actual file or database conversion. These are the benefits of this approach:

- Date Conversion
  - — Date conversion in current files, databases, and programs.
  - — Support of any date format that requires a century value for years ranging from 00 to 99.
  - — Compliance with ISO and ANSI date format standards.
- Support for a phased field expansion that does not require you to convert and migrate all impacted components at the same time.
- Minimal code change requirements, except for the bridging logic itself.

Using Bridge

Perform these tasks to streamline the application conversion process:

- Convert file record descriptions and change fields to the desired format.
- Build a Bridge Definition by using the Bridge interactive facility from the new record descriptions and generate the necessary Bridge Routines. Bridge uses the actual record description source in COBOL or PL/I. No special coding or calculations are required.
- Convert the programs, singly or in groups, to include the new record descriptions and program logic to correctly process the new fields. Add the Bridge CALL statement.
- Test the programs, comparing current output to output produced prior to the conversion.
- Promote the successfully tested converted programs into production.

After you have successfully converted all of the programs that access a file, you can convert the file itself. When you complete a file conversion, no program changes or recompiles are required. You can use the Bridge interactive facility to indicate that the file is converted and regenerate the Bridge Definition to disable bridging for that file. The Bridge CALL statement can be left in the program until all file conversions are completed and you determine that Bridge is no longer needed.

Application Knowledge Repository (AKR)

Bridge uses a special data store called an Application Knowledge Repository (AKR). At installation, you allocate a Bridge Management AKR to store installation and system level parameters and information about Generate Libraries associated with Bridge Definitions. You can also use the Bridge Management AKR to store your Bridge Definitions or to allocate separate AKRs.

Generate Library

A Generate Library is a load module library where Bridge stores the Bridge Routines generated from Bridge Definitions. Bridge tracks the Generate Libraries and members, and prevents Bridge Routines from being deleted or destroyed. You can store all Bridge Routines in a single Generate Library, or use multiple libraries.

Bridge Definition

Bridge stores information about a file and its bridging requirements in an object called a Bridge Definition. You must use one Bridge Definition for each file that requires bridging.

Attributes

Specify this information when you build a new Bridge Definition:

- Bridge Definition name.
- Program environment and attributes specific to the environment.
- AKR name where the Bridge Definition is stored.
- Group name. You can assign several Bridge Definitions to a group and then generate the Bridge Definitions by group.
- An optional long description of the Bridge Definition.
- DSN and member name of the COBOL or PL/I file description source.
- Language of the file description source (determined automatically).
- File conversion status, specifying whether file conversion is completed.
- DSN of the Generate Library where the generated Bridge Routine is stored.

Bridge Rules

Coded routines, called Bridge Macros, convert fields from one format to another. These macros are associated with the conversion fields in your record and are invoked by Bridge Rules. Bridge Rules identify the Bridge Macro and its associated parameters. Bridge comes with existing Bridge Rules that encompass most date formats and basic currency conversions. You can customize Bridge Rules by copying existing rules and modifying related Bridge Parameters.

If your application requires a field format that is not covered by the existing Bridge Macros, contact the ASG Service Desk.

Bridge Parameters

Bridge Parameters provide input to a Bridge Macro. You define Bridge Parameter values at the Bridge Definition level and override them at the Bridge Field level. You can also define your own parameters to create customized Bridge Rules (such as the transition to euro currency).

Note:

The Window Year attribute implemented in prior Bridge releases is now considered a parameter.

Record Definitions

Bridge uses Record Definitions to identify differing record formats within a file in order to correctly recognize and convert the appropriate fields. Record Definitions have one or more Field Tests you assign a name and test values to. You define Field Tests for each designated Test Field in the source record. Bridge provides an interactive facility that allows you to define and associate a Record Definition with every field within a record.

Generating Bridge Routines from Bridge Definitions

After you complete the Bridge Definitions, Bridge generates a load module that contains the Bridge Routine. The load module generates when you submit one or more batch jobs that analyze the record description in the original source format. Bridge then creates, assembles and link-edits the Bridge Routines. The Generate Library that is associated with the Bridge Definition stores the Bridge Routine. You can use the Bridge interactive facility to select the Bridge Definitions you want to generate by name, group, or environment in one or more AKRs.

Program Integration

To trigger Bridge, you add a single CALL to the initialization routine. This CALL associates the files you want processed by the program with the Bridge Routines generated from the Bridge Definitions.

After it is initialized, Bridge intercepts all I/O operations on a non-converted file and invokes the Bridge Routines to:

- Convert the records on input to the new format. The program receives the converted record.
- Convert the records on output to the old format. The program issues the WRITE command with a converted record.

Reverse Bridging

Occasionally, you might find it necessary to process a converted file with an unconverted program. For example, you converted the file to a new format (e.g., date fields expanded), but did not convert the program. The program expects to process an unconverted file. This is called reverse bridging, and is a feature of Bridge. You add a single CALL to the initializing routine to trigger reverse bridging.

Record-Level Application Programming Interface (RLAPI)

You can use the Bridge RLAPI feature when the standard run-time facility does not support the application or environment of the program you want to bridge, for example:

- Your program uses an access method that is not currently supported by Bridge, such as BSAM and EXCP.
- Your program runs in an environment not supported by Bridge, such as a commercial or home-grown database management system.
- You do not want to use the standard Bridge run-time engine.

See "Record-Level API" on page 173 for more detailed information about this feature.

3 Getting Started

This chapter describes getting started in Bridge and contains these sections:

This section explains how to perform these tasks:

- Initiate Bridge
- Use online help
- Verify or modify user options
- Import the Bridge Parameter and Bridge Rule starter sets
- Allocate or expand an AKR

If you are familiar with these tasks, you can proceed directly to "Building a Bridge Definition" on page 41.

Initiating Bridge

To initiate Bridge, use the ISPF selection or CLIST as indicated by your Systems Administrator. The method you use to invoke Bridge depends upon your system setup. If you require assistance, contact your System Administrator.

Online Help

Context sensitive online help answers most of your questions about Bridge screens and pop-ups. Help also includes information about abend messages and Bridge messages.

Screen and Pop-up Help

This manual describes all the options and fields for each screen and pop-up in detail, and notes any special processing considerations.

Abend Help

Online help lists and explains system and ESW abends.

Bridge Messages and User ABENDS

The Bridge runtime component issues messages that indicate the status of each file you want bridged and either the success or the failure of bridging. Each message is numbered and contains a severity level indicator. In addition, if Bridge encounters a severe problem, the program terminates with an abend code matching that of the message number. See "Program Messages and User Abends," on page 187 for a list of Bridge messages. These are the message severity levels:

Level	Description
I	Informational - Bridging is passing along an informational message.
W	Warning - Bridging detected an unusual condition and attempts to continue.
E	Error - Bridging encountered an error and cannot continue. The program is terminated. These errors are usually due to a problem that you can correct.
F	Fatal Error - Bridging encountered an internal error and is terminated (your program is also terminated). These are internal logic errors that you probably cannot correct.

Accessing Online Help

These are the methods to access online help:

Methods	Description
Press PF1 Type `HELP` or `?` on command line and press Enter	Displays help for the current screen or pop-up. If a message displays on the current screen or pop-up, help for that message displays.
Select Help on action bar	Help pull-down displays, listing a selection of help options. Select the option that fits your need.

Online Help Navigation Commands

These are the navigational commands for the help system after you access online help:

Command	Description
TOC	Goes to the Table of Contents.
INDEX	Goes to the help index. On an index screen, type a letter on the command line and press Enter to display index screen for that letter.
BACK	Goes to the previously displayed help screen.
Press Enter	Goes to the next screen in a continuation series.
HELP ABENDS	Goes to the Abends help screen where you can choose from a list of topics related to abends.
PF3 (END)	Exits online help.

Verifying User Options

At installation, Bridge sets the options that define the operating environment for interactive sessions to default values. The first time you use Bridge, verify that these options reflect the correct settings and change them if necessary:

User Options	Description
Product Parameters	Controls settings related to alarm, autosave, maximum Bridge Definition generations per job, and displays the Bridge Management AKR and Bridge Event Log name
Product Allocations	Specifies DASD information for the Log and List files
Log/List File Processing	Specifies the process for log/list files
PF Key Assignments	Specifies PF key assignments

You can access the user options from the Option pull-down on most screens or pop-ups by selecting an Option keyword on the action bar.

To verify or change product parameters

1 Select Option ▸ Product Parameters from the Bridge Primary screen (see Figure 7) and press Enter. The Options - Product Parameters pop-up displays (see Figure 8 on page 15).

Figure 7 • The ASG-Bridge Primary Screen

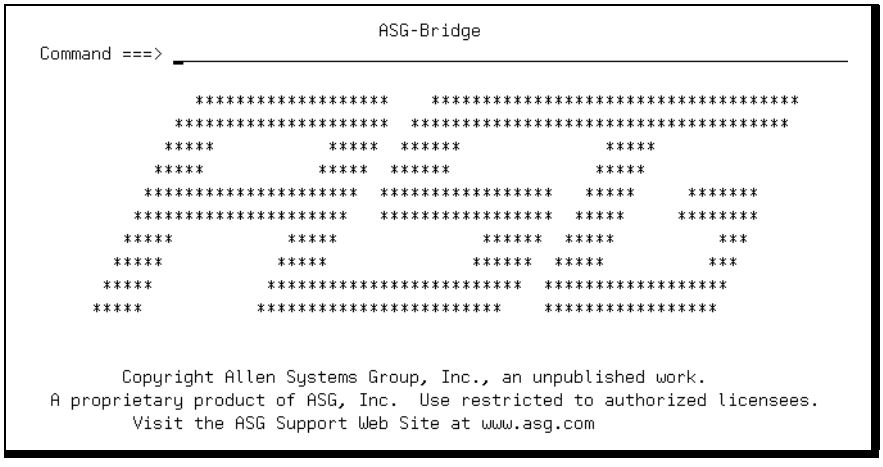
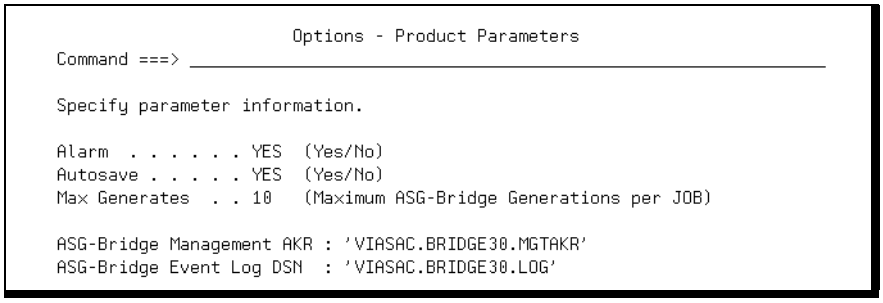


Figure 8 • Options - Product Parameters Pop-up



2 Verify or modify these parameters:

Parameter	Description
Alarm	YES for audible alarm or NO for no audible alarm when error message displays
Autosave	Specify YES for automatic save when exiting Bridge Definition screens Specify NO to receive a Save/Cancel prompt when exiting Bridge Definition screens
Max Generates	Maximum number of Bridge Definitions generated per job **Note:** Parameter is set at installation. Do not change without system administrator authorization
Bridge Management AKR	Display field showing name of base AKR set up during installation to manage Bridge information and operations You can store Bridge Definitions in this AKR
Bridge Event Log DSN	Display field showing name of Bridge Event Log specified during installation

3 Press PF3 to exit and save any changes.

To verify or change product allocations

1 Select Option ▸ Product Allocations from the Bridge Primary screen and press Enter. The Options - Product Allocations pop-up displays (see Figure 9).

Figure 9 • Options - Product Allocations Pop-up

```
                    Options - Product Allocations
Command ===> ____________________________________________________________

  Log file:
     Generic unit . . . SYSDA     (generic group name or unit address)
     Volume serial  . . ______    (blank for authorized default volume)
  List file:
     Generic unit . . . SYSDA     (generic group name or unit address)
     Volume serial  . . ______    (blank for authorized default volume)
```

2 Verify or modify the allocation information.

3 Press PF3 to exit and save any changes.

To verify or change Log/List options

1 Select Option ▸ Log/List from the Bridge Primary screen and press Enter. The Options - Log/List Definition pop-up displays (see Figure 10).

Figure 10 • Options - Log/List Definition Pop-up

```
                    Options - Log/List Definition
Command ===> ____________________________________________________________

1 - Process log file  2 - Process list file  3 - Customized data set name

Options                    Log           List
---------------            ---           ----
Process option  . . . . . . PD            PD
Primary tracks      . . . . 1             1
Secondary tracks    . . . . 2             5
Lines per page  . . . . . . 56            56
Sysout class  . . . . . . . *             *

Process options:  PK (print/keep), PD (print/delete), K, or D.


Job statement information:
  //NAME     JOB (ACCOUNT),NAME,
  //         MSGCLASS=A
  //*   INSERT '/*ROUTE PRINT NODE.USER' HERE IF NEEDED.
  //*
```

2 Verify or modify the process options.

3 If you specify the K or PK process option, you can customize the dataset where the log or list file is allocated.

a From the Options - Log/List Definition pop-up, type `3` and press Enter to display the Options - Log/List Name Customization pop-up shown in Figure 11.

By default, Bridge allocates the Log and List files as:

USERID.*pppnnnnn*.VIA*xxxxx*

where:

ppp is the product prefix.

nnnnn is a sequential number from 00001 to 99999.

xxxxx is LOG for Log and LIST for List.

b If you specify a TSO Prefix, the prefix appends to the beginning of the file name allocated for the Log and List files.

Figure 11 • Options - Log/List Name Customization Pop-up

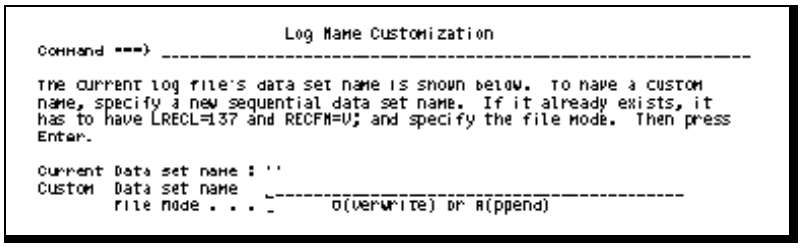
```
                Options - Log/List Name Customization
Command ===> ____________________________________________________________

You can define a customized name when you choose option PK (print/keep) or
K (keep) by specifying U(ser).  Specifying Y(es) on Prompt later lets you
define a custom name as you process each log/list file.  Otherwise,
specify data set name and file mode.  Then press Enter.

Options                  Log     List
------------------------ ---     ----
File Naming . . . . . . . S       S      U(ser) or S(ystem)
Prompt later for DSN  . . y       y      Y(es)  or N(o)

The following are needed if U(ser) and N(o) are specified above

Log Data set name   ____________________________________      (Seq)
     File Mode  . . _      O(verwrite) or A(ppend)
List Data set name  ____________________________________      (Seq)
     File Mode  . . _      O(verwrite) or A(ppend)
```

4 Type `U` in the File Naming field of the Options - Log/List Name Customization pop-up for Log and/or List to indicate a user-defined dataset name. If you specify `N` in the Prompt later for DSN field, you must enter a dataset name in the corresponding Data set name field, and specify Overwrite or Append in the File Mode field. Enter the appropriate information in these fields:

Field	Description
File Naming	Specifies whether the Log and List files are named by system default (S) or user-defined (U) dataset names
Prompt later for DSN	Enables you to define a custom name during processing If you specify `N`, you must enter a dataset name in the corresponding Data set name field, and specify Overwrite or Append in the File Mode field If you specify `Y` in the Prompt later for DSN field, Bridge prompts you for the dataset name during file processing
Log Data set name List Data set name	Indicates the name of the dataset where the Log or List files will be allocated during processing
File Mode	Specifies whether additional Log or List files overwrite or are appended to the existing files during processing

For example, if you specify `Yes` in the Prompt later for DSN field for the Log file, the Log Name Customization pop-up shown in Figure 12 displays.

Figure 12 • Log Name Customization Pop-up



5 Press PF3 to return to the Options - Log/List Definition screen.

6 Press PF3 to exit and save any changes.

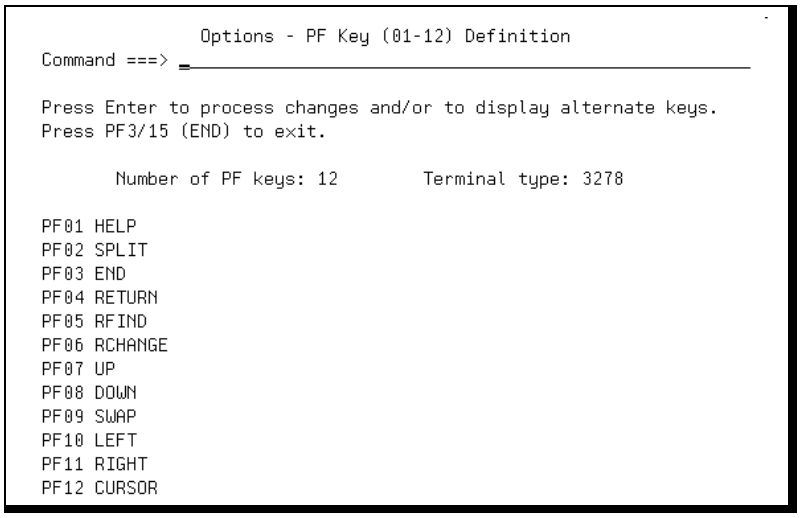
Note:

The Log file allocates only if an internal error, such as an abend, occurs. It contains ASG error messages that can aid the debugging process. The list file contains print output and is allocated when a print request is issued.

To verify or modify PF key assignments

1 Select Option ▸ PF Keys from the Bridge Primary screen and press Enter. The Options - PF Key Definition pop-up displays (see Figure 13).

Figure 13 • Options - PF Key Definition Pop-up with Defaults



2 Verify or modify PF key assignments. Type over a current PF key setting to change its function.

3 Press PF3 to exit and save any changes.

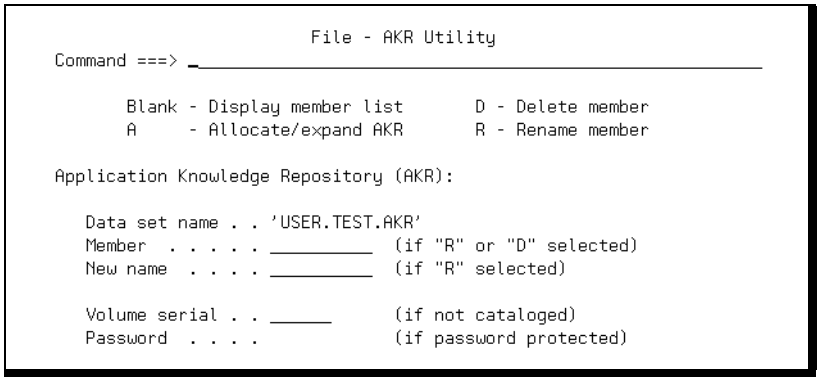
Allocating or Expanding an AKR

The AKR is a BDAM or VSAM file. When Bridge is installed it allocates a Bridge Management AKR to store installation parameters, system level parameters, and information about Generate Libraries. You can use this AKR to store your Bridge Definitions or you can allocate separate AKRs for Bridge Definition storage. Bridge includes AKR allocation and maintenance utilities.

To allocate or expand an AKR

1 Select File ▸ AKR Utility from the Bridge Primary screen and press Enter. The File - AKR Utility pop-up displays (see Figure 14 on page 20).

Figure 14 • File - AKR Utility Pop-up

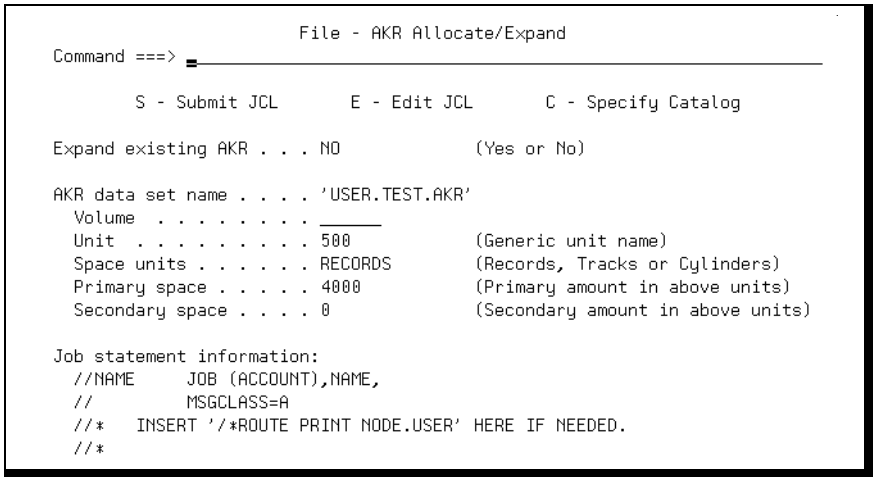


2 Type the necessary information in these fields:

Field	Description
Data set name	Name for the AKR If the high level qualifier is different than your TSO prefix or user ID, enclose the name in single quotes (')
Volume serial	Volume serial number where the AKR exists if the AKR is not cataloged
Password	Password if the AKR is password protected

3 Type `A` on the command line and press Enter. The File - AKR Allocate/Expand pop-up shown in Figure 15 displays.

Figure 15 • File - AKR Allocate/Expand Pop-up



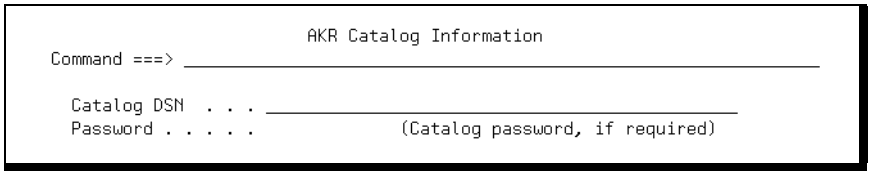
4 Type the necessary information in these fields:

Field	Description
Expand existing AKR	Select No to create a new AKR Select Yes to expand an existing AKR
Volume	AKR volume serial number
Unit	AKR generic unit name
Space units	Type of units to allocate: records, tracks, or cylinders
Primary Space	Amount of space required
Secondary Space	Amount of secondary space to allocate
Job statement information	Information necessary to complete the job statement

Note:

To specify a private catalog or password, type `C` on the command line and type the catalog dataset name and password on the AKR Catalog Information pop-up (see Figure 16).

Figure 16 • AKR Catalog Information Screen



5 Select one of these methods to submit the JCL to allocate the AKR:

Field	Description
Direct Submit	Type `S` on the command line and press Enter.
Edit and Submit	Type `E` on command line and press Enter to edit the JCL. Make your modifications and issue a standard TSO ISPF SUBMIT command. Press PF3 to return to the File - AKR Allocate/Expand pop-up.

6 Examine the job output to verify completion of the allocation job. To output displays condition codes of `0` if the allocation was successful.

Importing the Bridge Parameters and Bridge Rule Starter Sets

You define and maintain Bridge Parameters and Bridge Rules. However, to expedite and simplify your first use of the product, Bridge comes with a starter set of parameters and rules.

To employ these starter sets

1. Import the starter sets into Bridge as if you had previously exported them from Bridge.

2. Follow the instructions for importing Bridge Parameters and Bridge Rules in "Importing Bridge Parameters, Rules, or Definitions" on page 123 and import the starter sets of parameters and rules from these members in the ASG CNTL library.

Bridge Parameter Starter Sets

These are the Bridge parameter starter sets:

Member Name	Parameters for...
VIAGBPSS	Basic Bridge
VIAGYPSS	Bridge 2000
VIAGEPSS	Bridge for Euro

Bridge Rules Starter Sets

These are the Bridge rules starter sets:

Member Name	Parameters for...
VIAGBRSS	Basic Bridge
VIAGYRSS	Bridge 2000
VIAGERSS	Bridge for Euro

Note:

See your System Administrator or systems programmer who installed Bridge for the name of this dataset.

Migrating Bridge Definitions from Previous Versions of Bridge

If you are using a previous version of Bridge, you can migrate the Bridge Definitions from the AKRs built for that version to the Bridge 6.0 AKR. Using the previous version of Bridge, export the definitions you want to migrate to a sequential file and import them with Bridge 6.0.

Note:

IMS Bridge Definitions from Bridge Version 2.0 or earlier releases must be regenerated in Bridge 6.0. It is not necessary to regenerate non-IMS definitions.

After a program initializes bridging (through a call to VIAGFIMS or VIAGRIMS), it is in effect for all programs that run under the same TCB as the program that triggered Bridge. See "Integrating Bridge into IMS Programs" on page 148 for more information.

Customize the batch import process distributed with Bridge to invoke Bridge and import the definitions you exported to a sequential file. Use the VIABBTCJ member in the CNTL library to implement this process.

This is the syntax (that also displays at the end of the VIABBTCJ member) for customizing the ISPSTART entry that invokes Bridge:

```
//EXECESW.SYSTSIN DD *
  ISPSTRT PGM(VIAGMAIN)  NEWAPPL(VIAS)  PARM(RUNBATCH,  +
        INITCMD (BR_IMP_EXPORT OUTFILE_DSN=VIAXXX.YYYY.ZZZZ  +
                  MODE=IMPORT_REPLACE ENTITY=DCD  +
                  FILETYPE=SEQ AKR_DSN=VIASOFT.BRIDGExx.AKR))
```

These are the parameters you need to edit (you do not need to edit parameters that are not listed in the table, but you must include them):

Parameter	Specify...
OUTFILE_DSN	Dataset name containing exported Bridge Definitions
AKR_DSN	The AKR Bridge Definitions are imported into This AKR must be either blank (created with the AKR utility) or used only with the current version of Bridge.
MODE	IMPORT or IMPORT_REPLACE. The IMPORT_REPLACE option overwrites Bridge Definitions with the same name in the target AKR

Definitions containing an array that you created with previous versions of Bridge must be re-created in Bridge 6.0 before you generate a DCR (existing DCRs are not affected). You can either import or export the Bridge Definition, or use the RESYNCH option.

To resynch a Bridge 3.x definition

1 Select View ▸ ASG-Bridge Definition Directory from the Bridge Primary screen and press Enter. The View - ASG-Bridge Definition Directory screen (see Figure 17) displays.

Figure 17 • View - ASG-Bridge Definition Directory Screen

```
                    View - ASG-Bridge Definition Directory
Command ===> _____________________________________________ Scroll ===> CSR

AKR 'VIASAC.BRIDGE30.MGTAKR'                                      1 of 215

  A=Attributes    C=Change Environment   K=Check    D=Delete    E=Export
  V=Record View   I=ASG-Bridge Info      O=Copy     G=Generate  R=Rename

  Bridge   Env                                                          >
  Name     Type     Gen Record Layout Source                            >
- -------- -------- --- ------------------------------------------------->
_ A        IDMS     YES IDMS SCHEMA: EMPSCHM, VERSION: 300
_ AAAA     SEQ/VSAM NO  AUTOCHANGE: YYYY-RECORD(N/A)
_ ABC      IDMS     NO
_ ABCD     SEQ/VSAM NO  AUTOCHANGE: MASTER-IN(N/A)
_ ABCDE    SEQ/VSAM NO
_ ABCDEF   SEQ/VSAM NO
_ ABCDEFG  SEQ/VSAM NO
_ ACCTREC  SEQ/VSAM NO
_ ACTEST1  CICS     NO  AUTOCHANGE: WS-OUTLINE-OUTP(N/A)
_ ADVV415A CICS     YES VIASGS.USER.LIB(ADVV415A)
_ ATEST    SEQ/VSAM NO
```

2 Type V to the left of the Definition you want to view and press Enter to display the View Definition - Source Record screen.

3 Select File ▸ Edit Source. Press PF3 to display the Confirm Reload - Source Record pop-up (see Figure 18).

Figure 18 • Confirm Reload - Source Record Pop-up

```
                    Confirm Reload - Source Record
Command ===> ____________________________________________________

Press ENTER key to confirm reload request. Source will be resynched
if resynch option is selected

  Resynch Source  _

Enter END command to cancel reload request.
```

4 Select the Resynch Source Option, then press Enter to resynch the source code.

4 Creating Bridge Parameters and Bridge Rules

This chapter describes Bridge parameters and rules and contains these sections:

Bridge now supports your user-defined conversion efforts, such as date conversion and the transition to euro currency, by allowing you to create customized Bridge Rules with user-defined parameters.

Bridge Parameters

Bridge Parameters provide information to the Bridge Macro. You invoke the Bridge Macro by using a Bridge Rule. You define specific attributes to create the parameters which in turn define parameter usage and value. You can also review a list of existing Bridge Parameters, add, modify, delete, import, or export parameters.

Note:

For information on importing or exporting Bridge Parameters, see "Importing Bridge Parameters, Rules, or Definitions" on page 123.

Bridge Parameter Attributes

To modify Bridge parameter attributes

1 Select View ▸ ASG-Bridge Parameters from the Bridge Primary screen and press Enter. The View - ASG-Bridge Parameters screen displays (see Figure 19).

Figure 19 • View - ASG-Bridge Parameters Screen

```
                        View - ASG-Bridge Parameters
Command ===> _____________________________________________ Scroll ===> CSR

                                                              1 of 178

  C=Copy         D=Delete       E=Export       M=Modify       R=Rename

  Name     Keywrd Description                     Use Type Len Default Value >
- -------- ------ ------------------------------- --- ---- --- -------------->
_ ANYNEWS  NEWS   ANY NEWS SCALING FACTOR         VAL NUMB ANY 0
_ ANYOLDS  OLDS   ANY OLDS SCALING FACTOR         VAL NUMB ANY 0
_ ATESTB   ADJUST                                 OUT
_ ATEST1   DDMMYY TEST                            OUT
_ ATSEURO  EXCH   AUSTRIAN SCHILLING TO EURO CONVE VAL NUMB 7   1376030
_ AXEERATE EXCH   TEST EXCHANGE RATE              VAL NUMB 6   200000
_ BEFEURO  EXCH   BELGIAN FRANC TO EURO CONVERSION VAL NUMB 7   4033990
_ BRDGTSTA ADJUST TEST RULE                       OUT
_ CHECK    CHECK  CHECK NUMERIC VALUES            VAL CHAR ANY NO
_ CONV     NEWVAL NEW CURRENCY                    VAL CHAR 3   EUR
_ CONVERT  OLDVAL OLD  CURRENCY                   VAL ALPA ANY
_ CONVTYPE CONTYP CONVERSION TYPE,C->E,E->C,C->C  VAL CHAR 1
_ CURFLG   CURFLG CURRENCY TYPE FIELD             IN
```

2 Type `M` next to the name of the parameter you want to modify and press Enter. The View - Modify ASG-Bridge Parameter pop-up displays (see Figure 20).

Each parameter consists of attributes that establish data characteristics and default values.

Figure 20 • View - Modify ASG-Bridge Parameter Pop-up

```
                  View - Modify ASG-Bridge Parameter
Command ===> ____________________________________________________________

Modify ASG-Bridge Parameter information. Type CANCEL to cancel
modification.

Name  . . . . : ANYNEWS
Macro Keyword   NEWS
Description . . ANY NEWS SCALING FACTOR
Usage . . . . . V    ('I'nput field/'O'utput field/'V'alue)

The following information is needed if the Usage is Value only:

Value Type  . . N    ('C'haracter/'N'umber/'K'eyword/'A'lphaNumeric)
Value Length    ___  (a numeric value or blank -- not needed for Keyword)
Default Value   0
Allowed Values  ____________________________________________________
                ____________________________________________________
                ____________________________________________________
                ____________________________________________________
```

These are the parameter attributes:

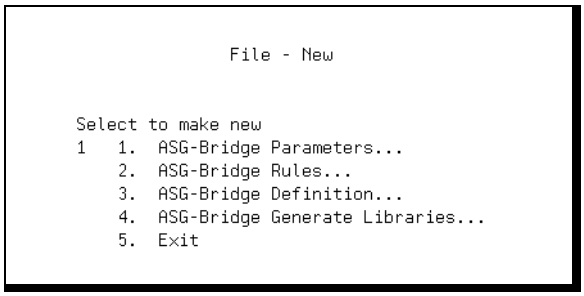
Attribute	Description
Name	Symbolic name referred to throughout the system
Macro Keyword	Keyword that displays in Bridge Rule Macro
Description	Brief description of the purpose of the parameter
Usage	Use parameters as input fields, output fields, or values
Input	A field in the record provides input to a Bridge Rule If you bridge a field within a VSAM key, the input parameter must also be within the key area
Output	Output provided as a result of the execution of a Bridge Rule Conversion Error might be considered an output parameter provided by a Bridge Rule that converts currency and reports a loss or gain as a conversion result If you bridge a field within a VSAM key, the output parameter must also be within the key area
Value	Specific value designations Value type and length, default value, and allowed values are characteristics specified to edit and validate parameter values Window Year or Currency Conversion Rate are examples of value parameters
Value Type	Type of data the parameter represents
Character	Value consists of alphabetic characters
Number	Value consists of numeric digits
Alphanumeric	Value consists of alphabetic, numeric, or a combination of alphabetic and numeric characters

Attribute	Description	
	Keyword	One or more of these pre-defined keywords: SPACE or SPACES LOW-VALUE or LOW-VALUES HIGH-VALUE or HIGH VALUES ZERO ZEROS ZEROES NULL NULLS QUOTE QUOTES
Value Length	Establishes a fixed length for the value	
Default Value	A specified default value A default value must meet the requirements of the specified Value Type and Value Length, and be one of the allowed values if allowed values are specified	
Allowed Values	Defines specifically allowed values, if any Specify one or more values, separating multiple values with commas	

To create a Bridge parameter

1 Select File ▸ New from the Bridge Primary screen and press Enter. The File - New pop-up displays (see Figure 21).

Figure 21 • File - New Pop-up



2 Select Bridge Parameters and press Enter. The File - New ASG-Bridge Parameter pop-up displays (see Figure 22).

Figure 22 • File - New ASG-Bridge Parameter Pop-up

```
                    File - New ASG-Bridge Parameter
Command ===> ____________________________________________________________

Enter ASG-Bridge Parameter information, then press Enter.

Name  . . . . . ________
Macro Keyword   ______
Description . . ____________________________
Usage . . . . . _    ('I'nput field/'O'utput field/'V'alue)

The following information is needed if the Usage is Value only:

Value Type  . . _    ('C'haracter/'N'umber/'K'eyword/'A'lphaNumeric)
Value Length    ___  (a numeric value or blank -- not needed for Keyword)
Default Value   ________________
Allowed Values  ____________________________________________________________
                ____________________________________________________________
                ____________________________________________________________
                ____________________________________________________________
```

3 Specify the required information and press Enter. A message displays verifying the Bridge Parameter is created.

4 Continue defining parameters or press PF3 to return to the File - New pop-up.

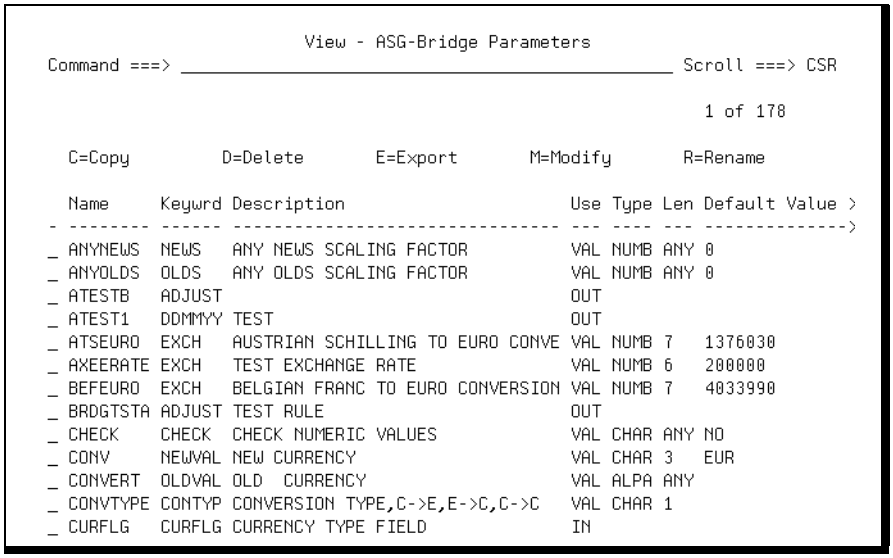
Viewing Existing Bridge Parameters

You can review the attributes of newly created or imported Bridge Parameters on the View - ASG-Bridge Parameters screen. This screen also provides actions to copy, delete, modify, export, or rename parameters.

To view existing Bridge Parameters

1 Select View ▸ ASG-Bridge Parameters from the Bridge Primary screen and press Enter. The View - ASG-Bridge Parameters screen displays (see Figure 23 on page 30).

Figure 23 • View - ASG-Bridge Parameters Screen



```
                          View - ASG-Bridge Parameters
Command ===> _____________________________________________ Scroll ===> CSR

                                                              1 of 178

  C=Copy         D=Delete       E=Export       M=Modify       R=Rename

  Name     Keywrd Description                       Use Type Len Default Value >
- -------- ------ --------------------------------- --- ---- --- --------------->
_ ANYNEWS  NEWS   ANY NEWS SCALING FACTOR           VAL NUMB ANY 0
_ ANYOLDS  OLDS   ANY OLDS SCALING FACTOR           VAL NUMB ANY 0
_ ATESTB   ADJUST                                   OUT
_ ATEST1   DDMMYY TEST                              OUT
_ ATSEURO  EXCH   AUSTRIAN SCHILLING TO EURO CONVE VAL NUMB 7   1376030
_ AXEERATE EXCH   TEST EXCHANGE RATE                VAL NUMB 6   200000
_ BEFEURO  EXCH   BELGIAN FRANC TO EURO CONVERSION  VAL NUMB 7   4033990
_ BRDGTSTA ADJUST TEST RULE                         OUT
_ CHECK    CHECK  CHECK NUMERIC VALUES              VAL CHAR ANY NO
_ CONV     NEWVAL NEW CURRENCY                      VAL CHAR 3   EUR
_ CONVERT  OLDVAL OLD  CURRENCY                     VAL ALPA ANY
_ CONVTYPE CONTYP CONVERSION TYPE,C->E,E->C,C->C    VAL CHAR 1
_ CURFLG   CURFLG CURRENCY TYPE FIELD               IN
```

2 Review the existing parameters or select an action to copy, delete, modify, export, or rename a parameter.

For more detailed instructions on copying, deleting, modifying, exporting, or renaming a parameter see these sections in this manual:

Action	See
Copy	"Copying Bridge Parameters or Bridge Rules" on page 34
Delete	"Deleting Bridge Parameters or Bridge Rules" on page 36
Modify	"Modifying Bridge Parameters or Bridge Rules" on page 37
Export	"Exporting Bridge Parameters, Rules, or Definitions" on page 116
Rename	"Renaming Bridge Parameters or Bridge Rules" on page 38

Overriding Bridge Parameter Defaults

After you define a new Bridge Parameter and associate it with a Bridge Rule and a Bridge Definition, you can override the initial values set for that parameter at either the Bridge Definition or Bridge Field level. These are the effects of the various override levels:

Level	Effect	Overrides
Parameter Definition (See "Creating Bridge Parameters and Bridge Rules" on page 25)	System-wide default	Nothing
Bridge Definition (See "Building a Bridge Definition" on page 41.)	Bridge Definition-wide default	System-wide default
Bridge Field (See "Defining Bridge Fields" on page 58.)	Single field default	Bridge and/or system defaults

Bridge Rules

Bridge Rules identify the Bridge Macro and its associated parameters. Bridge is distributed with the Adjust Rule that adjusts the length of the data item, and the Replace Rule which replaces the content in the data item. You can also create customized Bridge Rules either by associating existing rules with different Bridge Parameters, or by creating your own Bridge Macros and Parameters.

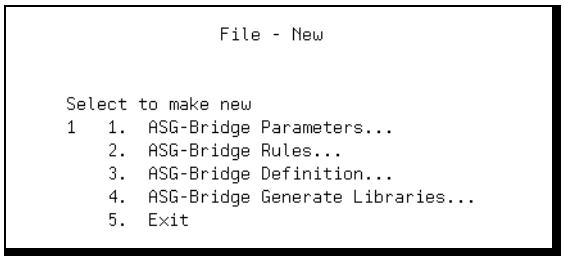
Note:

Depending on the type of Bridge you purchased, additional Bridge Rules may be available for date conversion and euro currency conversion.

To create a Bridge Rule

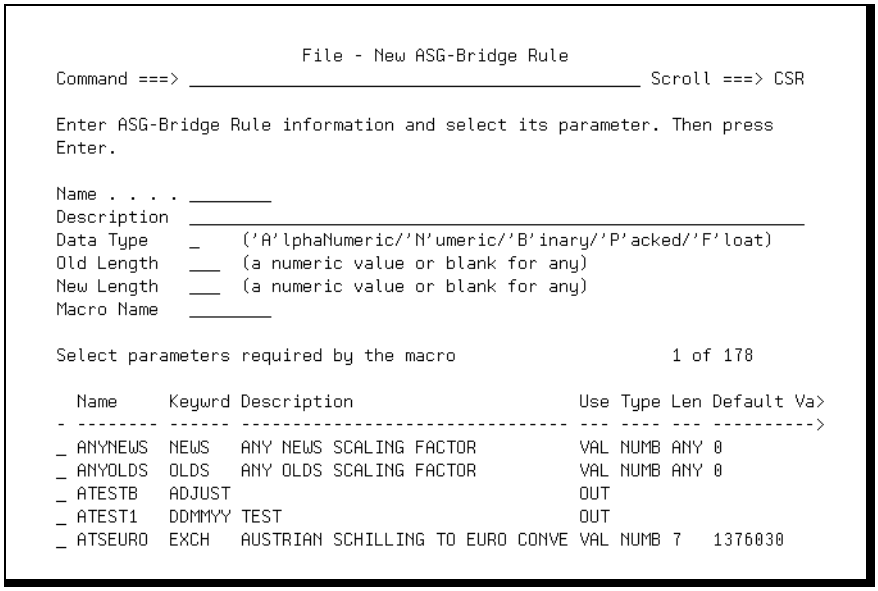
1 Select File ▸ New from the Bridge Primary screen and press Enter. The File - New pop-up (see Figure 24) displays.

Figure 24 • File - New Pop-up



2 Select ASG-Bridge Rules and press Enter. The File - New ASG-Bridge Rule pop-up displays (see Figure 25).

Figure 25 • File - New ASG-Bridge Rule Pop-up



```
                        File - New ASG-Bridge Rule
Command ===> ____________________________________________ Scroll ===> CSR

Enter ASG-Bridge Rule information and select its parameter. Then press
Enter.

Name . . . . ________
Description  ____________________________________________________________
Data Type    _    ('A'lphaNumeric/'N'umeric/'B'inary/'P'acked/'F'loat)
Old Length   ___  (a numeric value or blank for any)
New Length   ___  (a numeric value or blank for any)
Macro Name   ________

Select parameters required by the macro                    1 of 178

  Name     Keywrd Description                     Use Type Len Default Va>
- -------- ------ ------------------------------- --- ---- --- ---------->
_ ANYNEWS  NEWS   ANY NEWS SCALING FACTOR         VAL NUMB ANY 0
_ ANYOLDS  OLDS   ANY OLDS SCALING FACTOR         VAL NUMB ANY 0
_ ATESTB   ADJUST                                 OUT
_ ATEST1   DDMMYY TEST                            OUT
_ ATSEURO  EXCH   AUSTRIAN SCHILLING TO EURO CONVE VAL NUMB 7   1376030
```

3 Specify the required information in these fields and select any Bridge Parameters to associate with this Bridge Rule:

Field	Specify...
Name	Name for the new Bridge Rule
Description	Descriptive information related to Bridge Rule
Data Type	Whether data type is Alphanumeric, Numeric, Binary, Packed, or Float
Old Length	Previous field length or leave blank to indicate any length
New Length	New field length or leave blank to indicate any length
Macro Name	Name of Bridge Macro invoked by this Bridge Rule

Note:

See "Bridge Parameter Attributes" on page 26 for a description of the Bridge Parameter information shown in the Parameter Information field of the screen.

4 Press Enter. A message displays verifying that the Bridge Rule is created.

5 Continue defining rules or press PF3 to return to the File - New pop-up.

Viewing Existing Bridge Rules

You can review newly created or imported Bridge Rules on the View - ASG-Bridge Rules screen. This screen also allows you to copy, delete, modify, export, or rename rules.

To view existing Bridge Rules

1 Select View ▸ Bridge Rules from the Bridge Primary screen and press Enter. The View - ASG-Bridge Rules screen displays (see Figure 26).

Figure 26 • View - ASG-Bridge Rule Screen

```
                          View - ASG-Bridge Rules
Command ===> _____________________________________________ Scroll ===> CSR

                                                              1 of 174

  C=Copy        D=Delete       E=Export       M=Modify       R=Rename

                                                                      Data  >
  Name     Description                                                Type  >
- -------- ---------------------------------------------------------- ----- >
_ AA       PRACTICE                                                   ALPNM f
_ ACTACZ   NATIONAL CURRENCY TO NATIONAL CURRENCY - GENERAL RULE      NUM   f
_ ACTBOTH  ANY CURRENCY TO EURO PASS BACK INPUT VALUE                 NUM   f
_ ACTEUROZ ANY CURRENCY TO EURO ZONED WITH ROUNDING LOSS              NUM   f
_ ACTEURZN ANY CURRENCY TO EURO ZONED NO ROUNDING LOSS                NUM   f
_ ADJTEST  ADJUST THE SIZE OF A (FILLER) FIELD                        ALPNM f
_ ADJUST   ADJUST THE SIZE OF A (FILLER) FIELD                        ALPNM f
_ ADJUSTL  ADJUST SIZE OF FIELD TO THE LEFT                           ALPNM f
_ ADJUSTL1 SHIFT DATA 1 BYTE LEFT                                     NUM   f
_ ADJUSTN  ADJUST THE SIZE OF A (FILLER) FIELD                        NUM   f
_ ADJUSTNL ADJUST THE LENGTH OF NUMERIC FIELDS                        NUM   A
_ ANY2EURO ANY CURRENCY TO EURO                                       NUM   f_
```

2 Review the existing rules or select an action to copy, delete, modify, export, or rename a rule.

Copying Bridge Parameters or Bridge Rules

You can copy either an existing Bridge Parameter or Bridge Rule to create a new parameter or a new rule. You can then modify either the new parameter or the new rule if necessary.

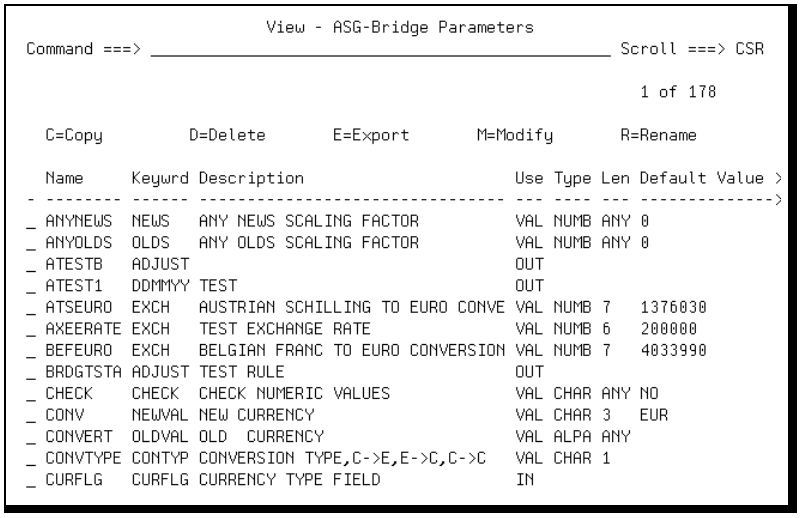
To copy a Bridge Parameter or Bridge Rule

1 Select View ▸ ASG-Bridge Parameters from the Bridge Primary screen and press Enter. The View - ASG-Bridge Parameters screen (see Figure 27).

Or

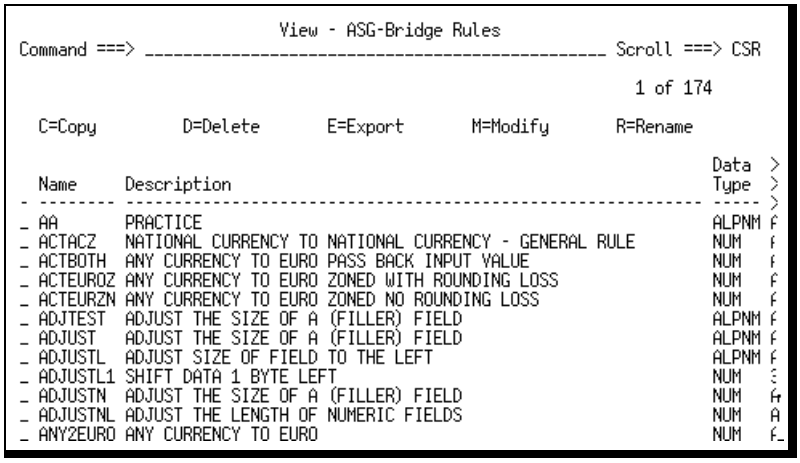
Select View ▸ ASG-Bridge Rules from the Bridge Primary screen and press Enter. The View - ASG-Bridge Rules screen (see Figure 28 on page 35) displays.

Figure 27 • View - ASG-Bridge Parameters Screen



```
                        View - ASG-Bridge Parameters
Command ===> _____________________________________________ Scroll ===> CSR

                                                              1 of 178

  C=Copy         D=Delete       E=Export       M=Modify       R=Rename

  Name     Keywrd Description                      Use Type Len Default Value >
- -------- ------ -------------------------------- --- ---- --- -------------->
_ ANYNEWS  NEWS   ANY NEWS SCALING FACTOR          VAL NUMB ANY 0
_ ANYOLDS  OLDS   ANY OLDS SCALING FACTOR          VAL NUMB ANY 0
_ ATESTB   ADJUST                                  OUT
_ ATEST1   DDMMYY TEST                             OUT
_ ATSEURO  EXCH   AUSTRIAN SCHILLING TO EURO CONVE VAL NUMB 7   1376030
_ AXEERATE EXCH   TEST EXCHANGE RATE               VAL NUMB 6   200000
_ BEFEURO  EXCH   BELGIAN FRANC TO EURO CONVERSION VAL NUMB 7   4033990
_ BRDGTSTA ADJUST TEST RULE                        OUT
_ CHECK    CHECK  CHECK NUMERIC VALUES             VAL CHAR ANY NO
_ CONV     NEWVAL NEW CURRENCY                     VAL CHAR 3   EUR
_ CONVERT  OLDVAL OLD  CURRENCY                    VAL ALPA ANY
_ CONVTYPE CONTYP CONVERSION TYPE,C->E,E->C,C->C   VAL CHAR 1
_ CURFLG   CURFLG CURRENCY TYPE FIELD              IN
```

Figure 28 • View - ASG-Bridge Rule Screen



2 Type C to select either the parameter or the rule to copy and press Enter. The ASG-Bridge Parameter Copy (see Figure 29) or the ASG-Bridge Rule Copy (see Figure 30) pop-up displays.

Figure 29 • ASG-Bridge Parameter Copy Pop-up

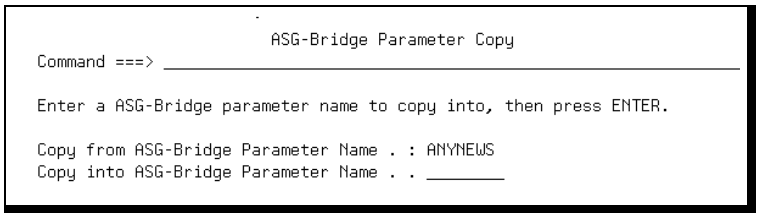
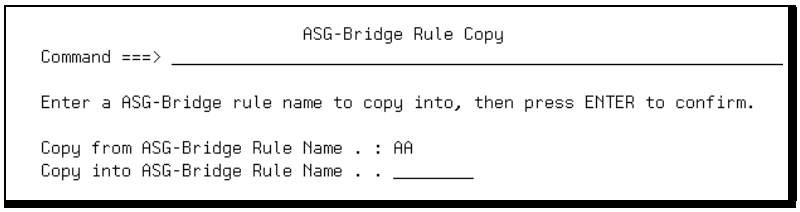


Figure 30 • ASG-Bridge Rule Copy Pop-up



3 Specify either the copy into parameter or the rule name and press Enter. A message displays on either the View - ASG-Bridge Parameters or the View - ASG-Bridge Rules screen verifying that the new parameter or the new rule is added.

Deleting Bridge Parameters or Bridge Rules

You can delete either Bridge Parameters or Rules that are no longer needed. However, if a parameter is in use, you cannot delete it. If a rule is in use, you can delete it, but you will not be able to generate any Bridge Definitions that used that rule. ASG does not recommend you delete Bridge Rules that are in use.

To delete a Bridge Parameter or Bridge Rule

1 Select View ▸ ASG-Bridge Parameters from the Bridge Primary screen. The View - ASG-Bridge Parameters screen displays.

Or

Select View ▸ ASG-Bridge Rules screen from the Bridge Primary screen. The View - ASG-Bridge Rules screen displays.

2 Type `D` to select the parameters or rules to delete and press Enter. The Confirm Delete - ASG-Bridge Parameter (see Figure 31) or the Confirm Delete - ASG-Bridge Rule (see Figure 32) pop-up displays.

Figure 31 • Confirm Delete - ASG-Bridge Parameter Pop-up

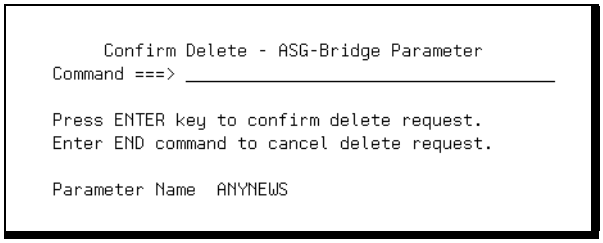
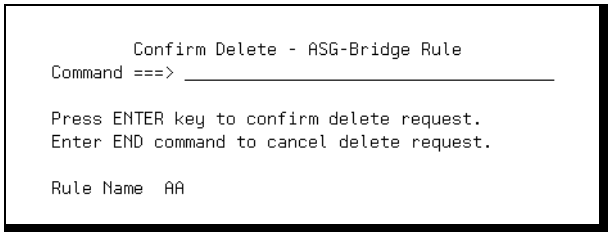


Figure 32 • Confirm Delete - ASG-Bridge Rule Pop-up



3 Press Enter to confirm the deletion.

Or

Press PF3 to cancel the delete request. A message displays on the View - ASG-Bridge Parameters or on the View - ASG-Bridge Rules screen either verifying your action or advising why the deletion cannot occur.

Modifying Bridge Parameters or Bridge Rules

You can modify existing Bridge Parameters or Bridge Rules if you need to change or correct the information.

Note:

ASG recommends that you do not modify parameters or rules that are in use as you may receive unpredictable results.

To modify Bridge Parameters or Bridge Rules

1 Select View ▸ ASG-Bridge Parameters from the Bridge Primary screen and press Enter. The View - ASG-Bridge Parameters screen displays.

Or

Select View ▸ ASG-Bridge Rules from the Bridge Primary screen and press Enter. The View - ASG-Bridge Rules screen displays

2 Type `M` to select the parameter or rule to modify and press Enter. The View - Modify ASG-Bridge Parameter (see Figure 33) or View - Modify ASG-Bridge Rule (see Figure 34 on page 38) pop-up displays.

Figure 33 • View - Modify ASG-Bridge Parameter Screen

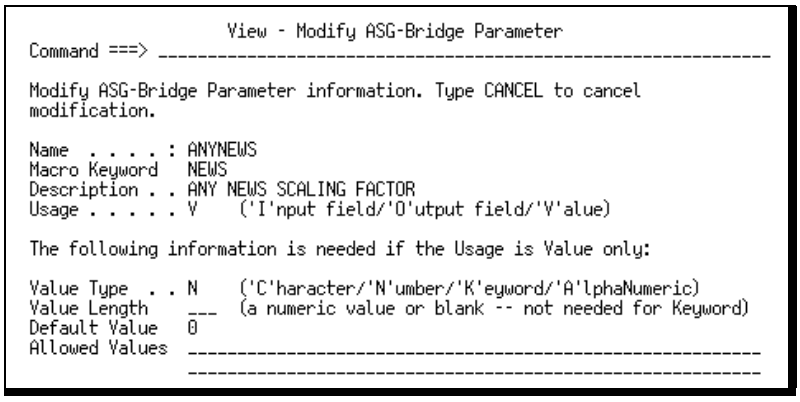
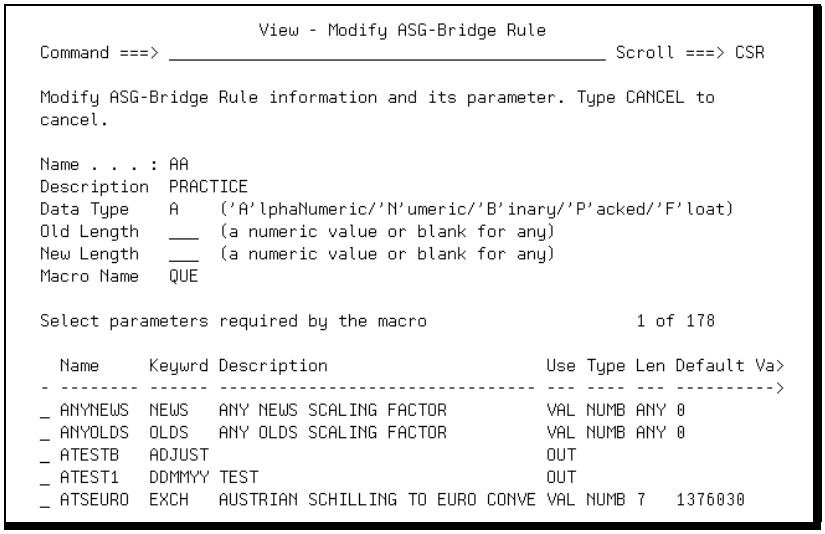


Figure 34 • View Modify ASG-Bridge Rule Pop-up



```
                    View - Modify ASG-Bridge Rule
Command ===> _____________________________________________ Scroll ===> CSR

Modify ASG-Bridge Rule information and its parameter. Type CANCEL to
cancel.

Name . . . : AA
Description  PRACTICE
Data Type    A    ('A'lphaNumeric/'N'umeric/'B'inary/'P'acked/'F'loat)
Old Length   ___  (a numeric value or blank for any)
New Length   ___  (a numeric value or blank for any)
Macro Name   QUE

Select parameters required by the macro                    1 of 178

  Name     Keywrd Description                      Use Type Len Default Va>
- -------- ------ -------------------------------- --- ---- --- ----------->
_ ANYNEWS  NEWS   ANY NEWS SCALING FACTOR          VAL NUMB ANY 0
_ ANYOLDS  OLDS   ANY OLDS SCALING FACTOR          VAL NUMB ANY 0
_ ATESTB   ADJUST                                  OUT
_ ATEST1   DDMMYY TEST                             OUT
_ ATSEURO  EXCH   AUSTRIAN SCHILLING TO EURO CONVE VAL NUMB 7   1376030
```

3 Make the necessary changes and press PF3 to save the changes and return to the View - ASG-Bridge Parameters or the View - ASG-Bridge Rules screen. A message displays on the View - ASG-Bridge Parameters or the View - ASG-Bridge Rules screen verifying that the either new parameter or the new rule is modified.

Note:

Type `cancel` on the command line of the View - Modify ASG-Bridge Parameters or the View - Modify ASG-Bridge Rules pop-up to cancel changes and return to the View - ASG-Bridge Parameters or the View - ASG-Bridge Rules screen.

Renaming Bridge Parameters or Bridge Rules

You can rename existing Bridge Parameters or Bridge Rules. You cannot rename a Bridge Parameter if it is in use. You can rename a Bridge Rule that is in use, but you will not be able to generate any Bridge Definitions that use the old name. ASG does not recommend renaming Bridge Rules that are in use.

To rename Bridge Parameters or Bridge Rules

1 Select View ▸ ASG-Bridge Parameters. The View - ASG-Bridge Parameters screen displays.

Or

Select View ▸ ASG-Bridge Rules screen. The View - ASG-Bridge Rules screen displays.

2 Type `R` to select the parameter or rule to rename. Either the ASG-Bridge Parameter Rename (see Figure 35) or the ASG-Bridge Rule Rename (see Figure 36) pop-up displays.

Figure 35 • ASG-Bridge Parameter Rename Pop-up

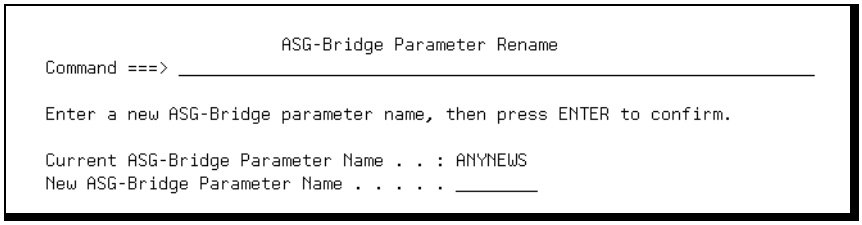
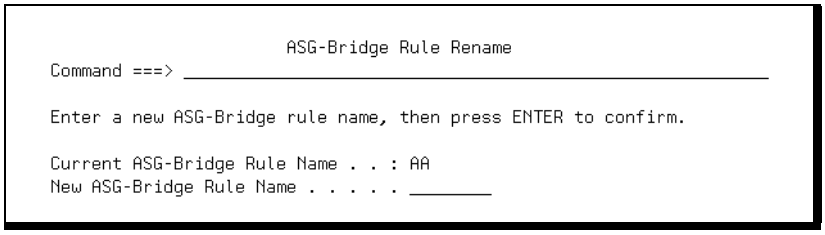


Figure 36 • ASG-Bridge Rule Rename Pop-up



3 Specify the new parameter or the new rule name and press Enter to save the new name and return to the View - ASG-Bridge Parameters or the View - ASG-Bridge Rules screen. A message displays on either the View - ASG-Bridge Parameters or the View - ASG-Bridge Rules screen verifying that the parameter or rule is renamed.

5 Building a Bridge Definition

This chapter describes how to build a Bridge definition and contains these sections:

You need one Bridge Definition for each file you want bridged. This is the information the Bridge Definition identifies:

- Program environment
- The expanded record
- All record formats in the file
- Fields you want to convert and their respective Bridge Rules
- Arrays occurring in the record
- Padding fields
- Primary and alternate record keys

Name and Attributes of a Bridge Definition

To begin building a new Bridge Definition, name it and specify environment, source, storage, and file conversion attributes. These attributes tell Bridge the program environment, where to find the source file, where to store the generated Bridge Routine information, and how to interpret the values that you want bridged.

Bridge Definition Name

The Bridge Definition name becomes the name of an executable Bridge Routine. When naming the Bridge Definition, use a legal program name with an alphabetic first character. The remaining characters can be alphanumeric. You can also rename an existing Bridge Definition (see "Renaming an Existing Bridge Definition" on page 84).

When you name the Bridge Definition, you can also designate a group name and a long description. Several Bridge Definitions can belong to the same group. This allows you to generate executable Bridge Routines for all the Bridge Definitions within a group (see "Generating Executable Bridge Routines" on page 95). You can modify the group description and long description of an existing Bridge Definition (see "Modifying Bridge Definition Group and Description" on page 85).

Bridge Definition Attributes

After naming the Bridge Definition, you can specify Bridge Definition attributes. Some attributes display only for the designated environment. You can modify the attributes of an existing Bridge Definition (see "Modifying Bridge Definition Attributes" on page 87). These are the attributes you can specify:

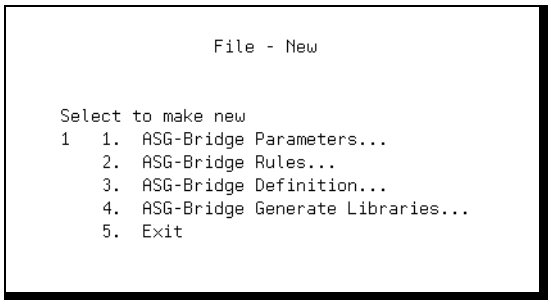
Attribute	Description	Environment
AKR	AKR name that stores Bridge Definition	All
Bridge Name	Bridge Definition name	All
Alias Name	Secondary name for executable Bridge Routine	CICS DL/I
Source	Expanded record description source	All
Language	Language of expanded record description source Display field only (Bridge automatically determines the language)	All
File Conversion Completed	File conversion status `No` indicates conversion not complete `Yes` indicates conversion is complete.	Sequential/VSAM CICS
DBD Conversion Completed	Database description conversion status `No` indicates conversion not complete `Yes` indicates conversion is complete.	IMS

Attribute	Description	Environment
IMS GSAM	Yes indicates GSAM access method No indicates non-GSAM access method	IMS
Record format	Specifies whether record format is Fixed (F) or Variable (V)	All
Record Length (LRECL)	Record length Specify for fixed record format only Enter the unexpanded record length of the physical file.	CICS
Old DBD Library	Name of original database description library	IMS
New DBD Library	Name of converted database description library	IMS
Generate Library	Generate Library where Bridge Routine generated from this definition is stored	All

To name a new Bridge Definition and specify definition attributes

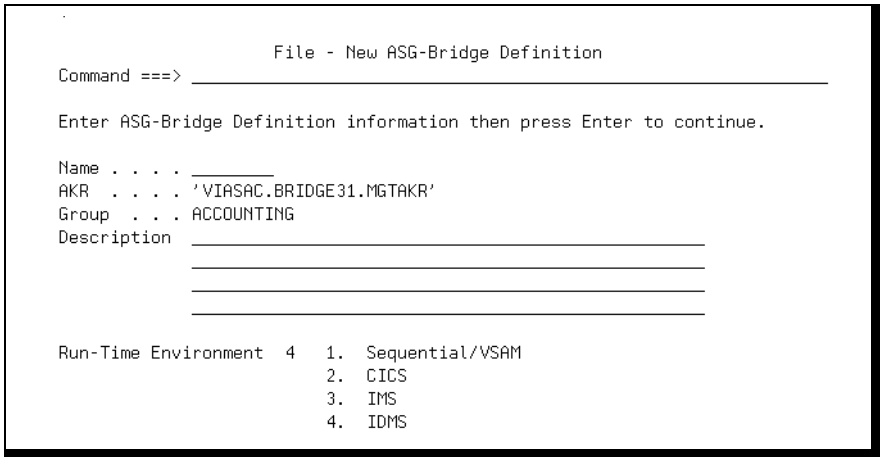
1 Select File ▸ New from the Bridge Primary screen and press Enter. The File - New pop-up displays (see Figure 37).

Figure 37 • File - New Pop-up



2 Select ASG-Bridge Definition and press Enter.
The File - New ASG-Bridge Definition pop-up displays (see Figure 38 on page 44).

Figure 38 • File - New ASG-Bridge Definition Pop-up



3 Type the name, AKR, Group, and Description information in the appropriate fields and select an environment.

Bridge Definitions for CICS and Sequential/VSAM are interchangeable. Use CICS to specify the Record Format and the LRECL of the file. This information is required for CICS remote files where the local FCT does not contain the information or for PL/I access to files with RECFM=F in batch.

4 Press Enter. The ASG-Bridge Definition - Attributes screen displays for one of these environments:

- Sequential/VSAM Environment (see Figure 39)
- CICS Environment (see Figure 40 on page 45)
- AIMS CICS DL/I Environment (see Figure 41 on page 45)
- IDMS Environment (see Figure 42 on page 45)

Figure 39 • ASG-Bridge Definition - Attributes Screen (Sequential/VSAM Environment)

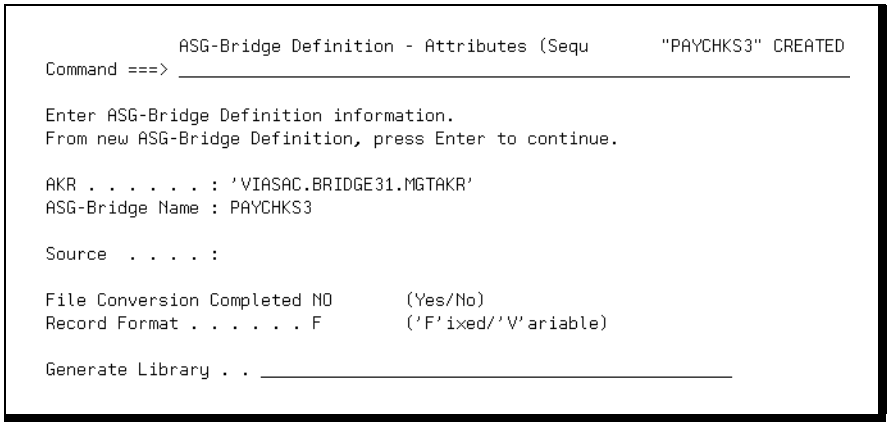


Figure 40 • ASG-Bridge Definition - Attributes Screen (CICS Environment)

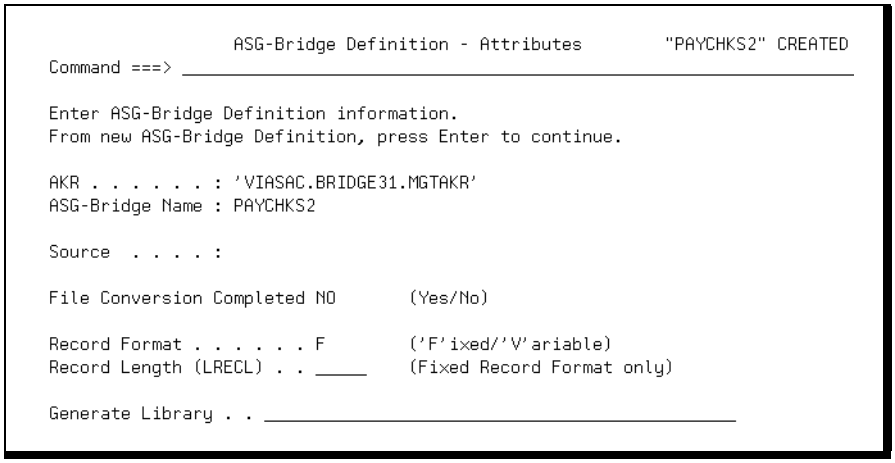


Figure 41 • ASG-Bridge Definition - Attributes Screen (IMS CICS DL/I Environment)

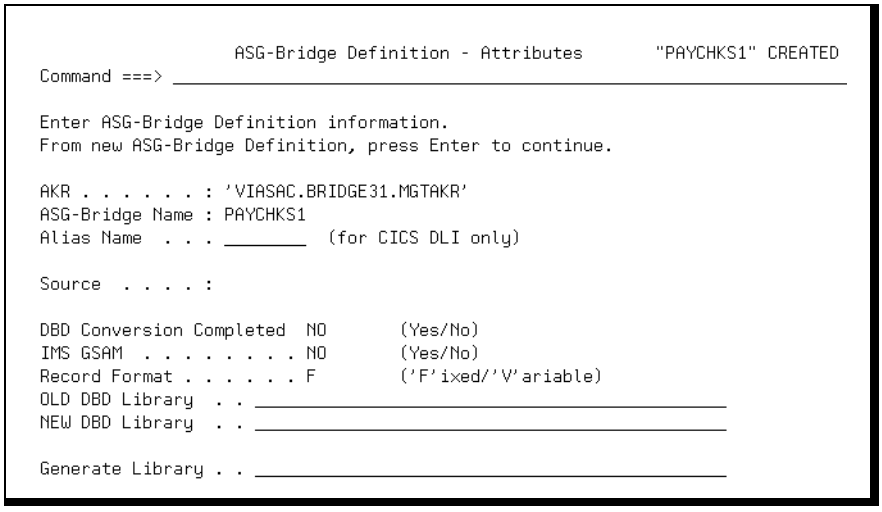
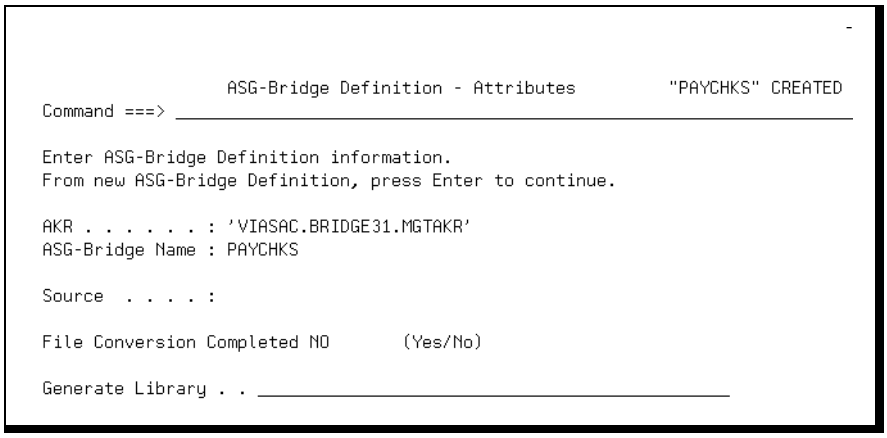


Figure 42 • ASG-Bridge Definition - Attributes Screen (IDMS Environment)



5 On the ASG-Bridge Definition - Attributes pop-up, specify the environment and conversion attributes for the selected environment.

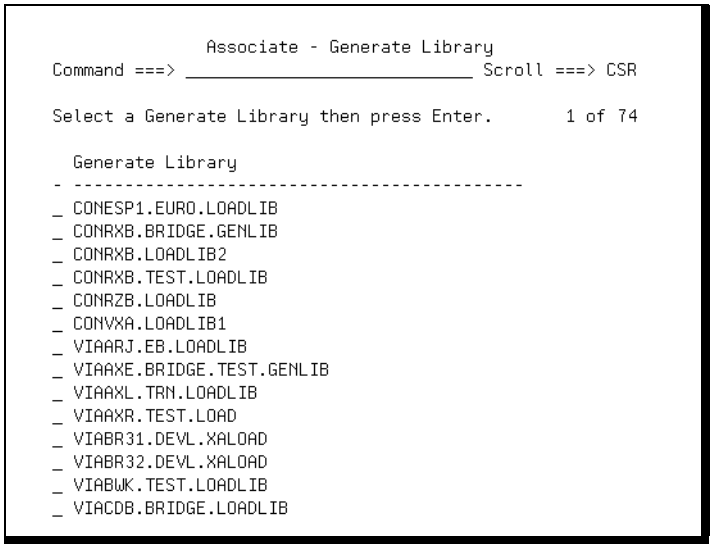
6 To specify the name of a Generate Library to associate with the Bridge Definition, type the Generate Library name in the Generate Library field.

Or

Select File ▸ Select Generate Library.

The Associate - Generate Library pop-up displays (see Figure 43).

Figure 43 • Associate - Generate Library Pop-up



7 Select a Generate Library from the list on the Associate - Generate Library pop-up.

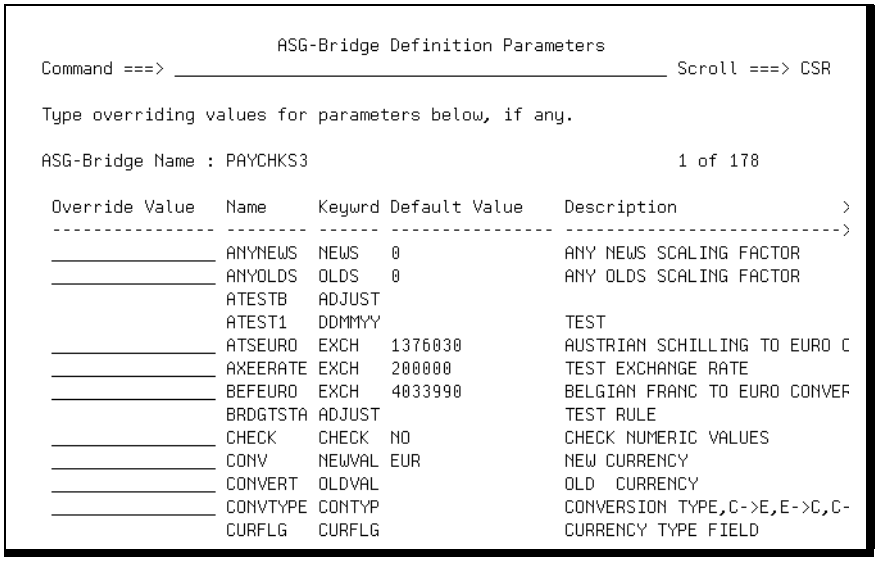
8 Press Enter to specify the selected library and return to the Bridge Definition - Attributes screen.

Note:

If you did not specify a Bridge Generate Library to Bridge, see "Maintaining Generate Libraries" on page 104.

9 Select Edit ▸ Bridge Parameter to specify any necessary Bridge Parameter overrides. The Bridge Definition Parameters screen displays (Figure 44 on page 47).

Figure 44 • ASG-Bridge Definition Parameters Screen

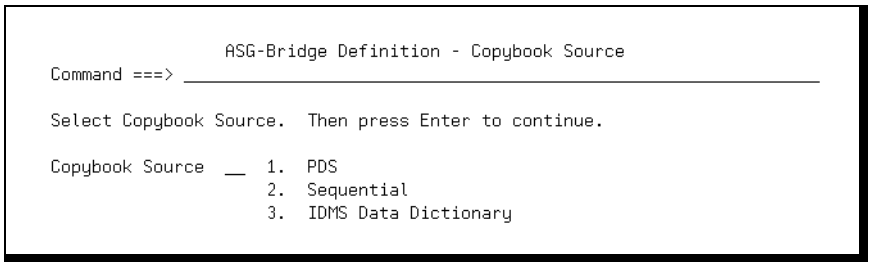


```
                    ASG-Bridge Definition Parameters
Command ===> _____________________________________________ Scroll ===> CSR

Type overriding values for parameters below, if any.

ASG-Bridge Name : PAYCHKS3                                    1 of 178

 Override Value   Name     Keywrd Default Value    Description               >
 ---------------- -------- ------ ---------------- --------------------------->
 _______________  ANYNEWS  NEWS   0                ANY NEWS SCALING FACTOR
 _______________  ANYOLDS  OLDS   0                ANY OLDS SCALING FACTOR
                  ATESTB   ADJUST
                  ATEST1   DDMMYY                  TEST
 _______________  ATSEURO  EXCH   1376030          AUSTRIAN SCHILLING TO EURO C
 _______________  AXEERATE EXCH   200000           TEST EXCHANGE RATE
 _______________  BEFEURO  EXCH   4033990          BELGIAN FRANC TO EURO CONVER
                  BRDGTSTA ADJUST                  TEST RULE
 _______________  CHECK    CHECK  NO               CHECK NUMERIC VALUES
 _______________  CONV     NEWVAL EUR              NEW CURRENCY
 _______________  CONVERT  OLDVAL                  OLD  CURRENCY
 _______________  CONVTYPE CONTYP                  CONVERSION TYPE,C->E,E->C,C-
                  CURFLG   CURFLG                  CURRENCY TYPE FIELD
```

10 Specify the necessary parameter override values in the Override Value field and press PF3 to save and return to the ASG-Bridge Definition - Attributes screen.

11 Press Enter to specify source information. The ASG-Bridge Definition - Copybook Source screen shown in Figure 45 displays.

Figure 45 • ASG-Bridge Definition - Copybook Source Pop-up



```
                ASG-Bridge Definition - Copybook Source
Command ===> ____________________________________________________________

Select Copybook Source.  Then press Enter to continue.

Copybook Source  __  1.  PDS
                     2.  Sequential
                     3.  IDMS Data Dictionary
```

Note:

If your run time is IDMS, the Bridge Definition - IDMS Information screen displays when you press Enter (see Figure 48 on page 49).

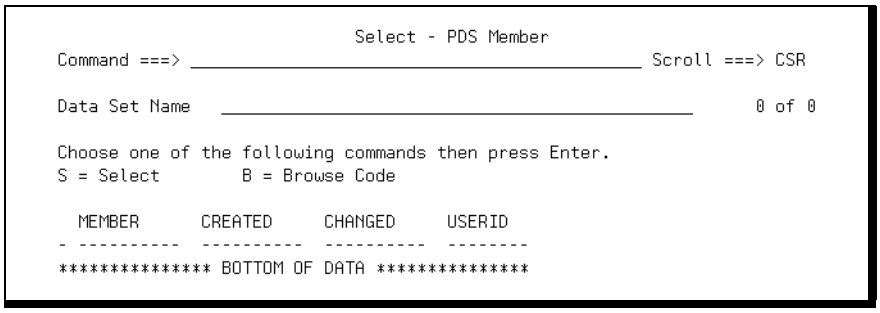
a Select option 1- PDS and press Enter to perform one of these actions. The Select - PDS Member screen displays (see Figure 46). Press PF3 to return to the Select - PDS Member pop-up.

— Type the name of the dataset and press Enter.

— To load the source, use the S action to select a member and press Enter.

— To browse the member source code before selecting it, use the B action.

Figure 46 • Select - PDS Member Pop-up

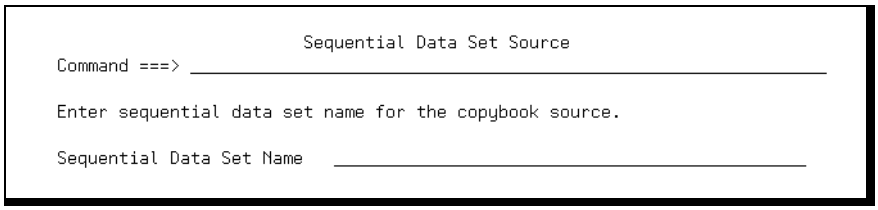


b Select option 2 - Sequential and press Enter to perform one of these actions. The Sequential Data Set Source pop-up displays (see Figure 47).

— To verify the source, type the name of the dataset and press Enter.

— To exit and view the source, press PF3.

Figure 47 • Sequential Data Set Source Pop-up



c Select option 3 - IDMS Data Dictionary and press Enter to perform one of these actions. The ASG-Bridge Definition - IDMS Information pop-up displays (see Figure 48 on page 49).

— To specify non-IDMS records stored in a Data Dictionary, select the IDMS Data Dictionary Information option. The ASG-Bridge Definition - IDMS Information pop-up (Figure 48 on page 49) displays.

— Specify any IDMS parameters and DD information to properly invoke IDMS. Use the I or D action to insert or delete DDs. Type NEXT on the command line and press Enter. The ASG-Bridge Definition - Non-IDMS Record Names pop-up displays (see Figure 49).

— Select a record and press PF3 to load the source.

Figure 48 • ASG-Bridge Definition - IDMS Information (non-IDMS Definition)

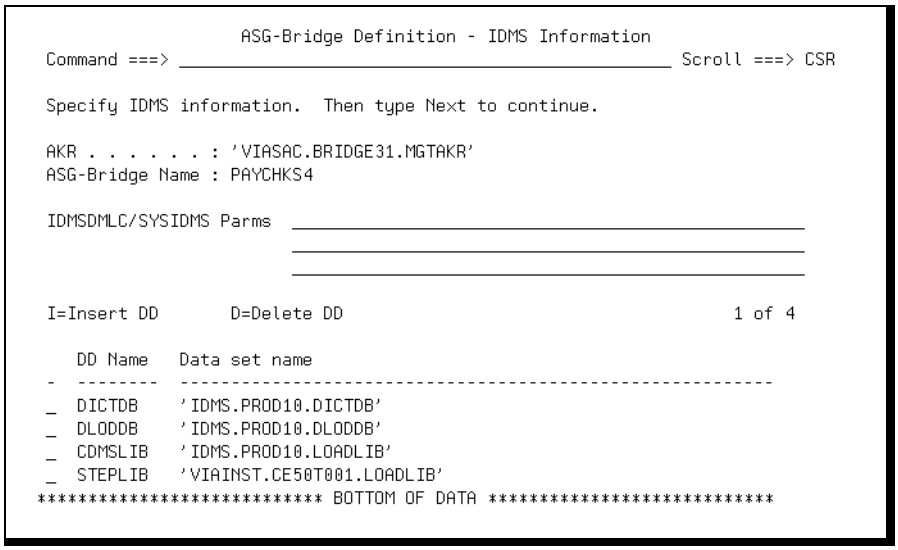
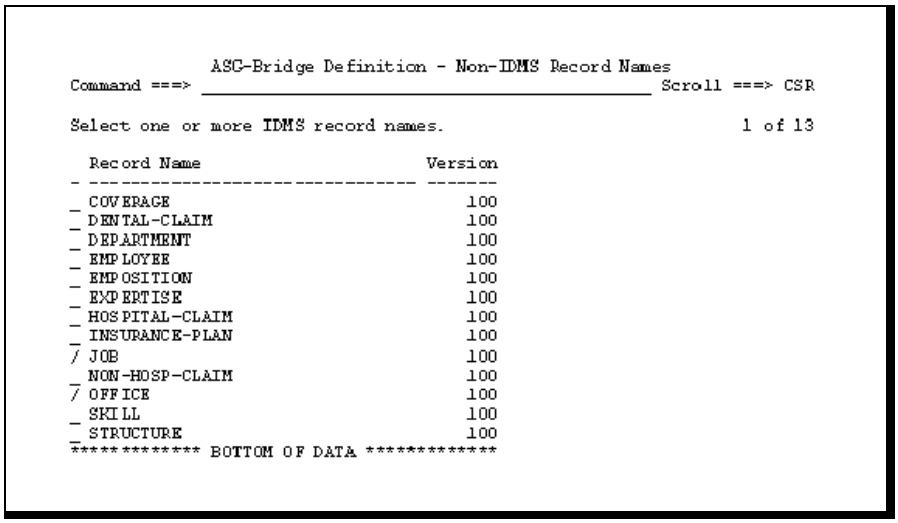


Figure 49 • ASG-Bridge Definition - Non-IDMS Record Names



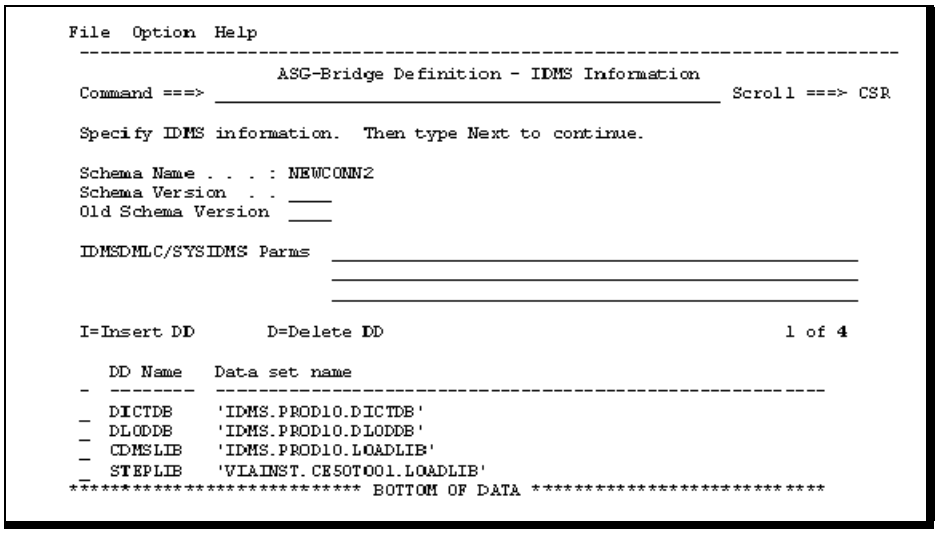
To create a IDMS Bridge definition

Note:

The Bridge IDMS option must be installed to complete this step.

1 Select 3. IDMS Data Dictionary from the Bridge Definition - Copybook Source screen shown in Figure 45 on page 47. Press enter to display the Bridge Definition - IDMS Information pop-up (see Figure 50).

Figure 50 • ASG-Bridge Definition - IDMS Information (IDMS Definition)

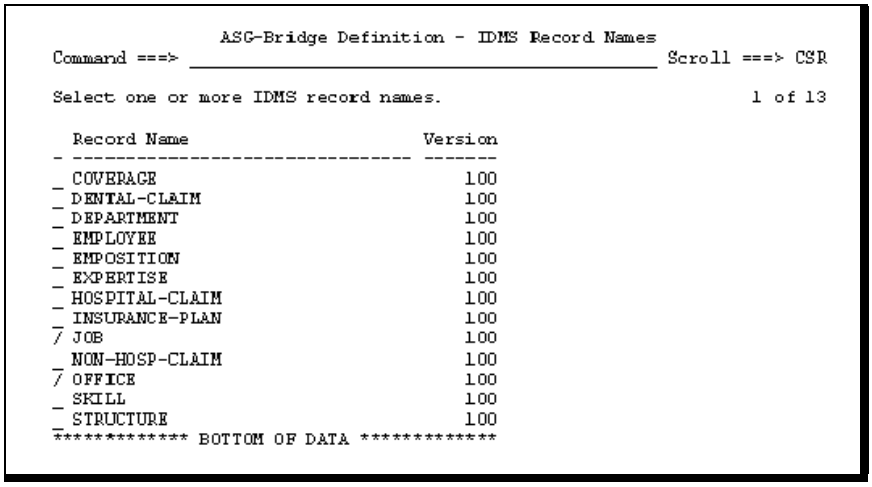


2 Specify any IDMS parameters and DD information to properly invoke IDMS. Use either the I action to insert DDs or the D action to delete DDs.

3 Type `NEXT` on the command line. The ASG-Bridge Definition - IDMS Record Names pop-up displays (Figure 51 on page 51).

4 Select one or more records and press PF3 to load the source.

Figure 51 • ASG-Bridge Definition - IDMS Record Names



```
                ASG-Bridge Definition - IDMS Record Names
Command ===> _____________________________________________ Scroll ===> CSR

Select one or more IDMS record names.                              1 of 13

  Record Name                      Version
- -------------------------------- -------
_ COVERAGE                            100
_ DENTAL-CLAIM                        100
_ DEPARTMENT                          100
_ EMPLOYEE                            100
_ EMPOSITION                          100
_ EXPERTISE                           100
_ HOSPITAL-CLAIM                      100
_ INSURANCE-PLAN                      100
/ JOB                                 100
_ NON-HOSP-CLAIM                      100
/ OFFICE                              100
_ SKILL                               100
_ STRUCTURE                           100
************* BOTTOM OF DATA *************
```

5 The ASG-Bridge Definition - Source Record screen displays for these different environments:

- Sequential/VSAM and CICS (see Figure 52 on page 52)
- IMS/Non-GSAM (see Figure 53 on page 52)
- IMS/GSAM (see Figure 54 on page 52)
- IDMS (see Figure 55 on page 53)

6 To define Bridge Fields and Record Definitions, proceed to the next section, "Defining Source Record Information" on page 54.

Note:

The actions available on each ASG-Bridge Definition - Source Record screen vary according to the environment. After you create a Bridge Definition, it is accessible from the View - ASG-Bridge Definition Directory screen. See "Bridge Definition Directory" on page 80 for more information.

Figure 52 • ASG-Bridge Definition - Source Record Screen (SEQ/VSAM and CICS)

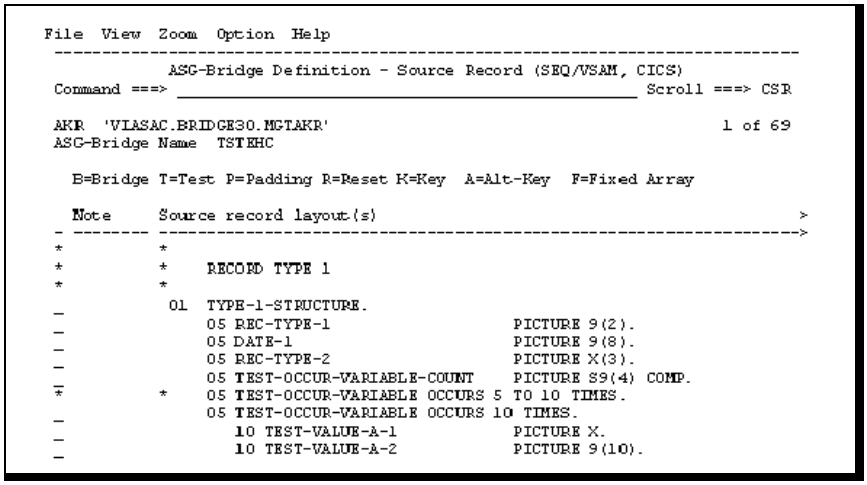


Figure 53 • ASG-Bridge Definition - Source Record Screen (IMS/Non-GSAM)

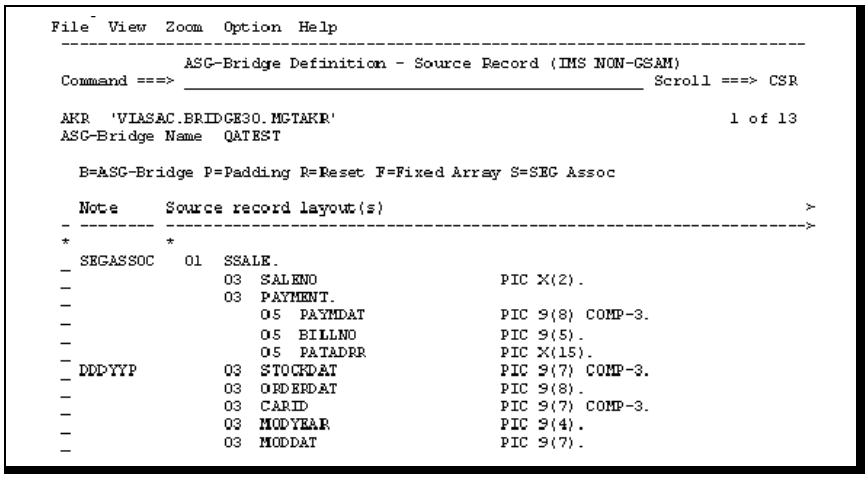


Figure 54 • ASG-Bridge Definition - Source Record Screen (IMS GSAM)

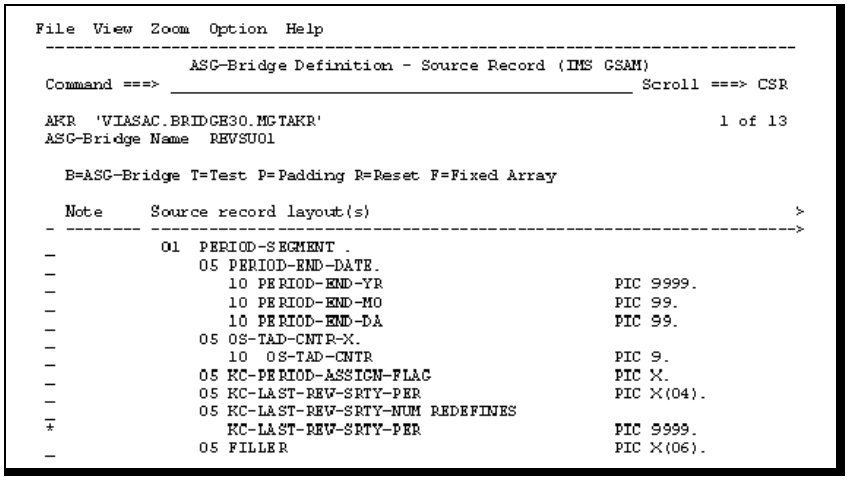
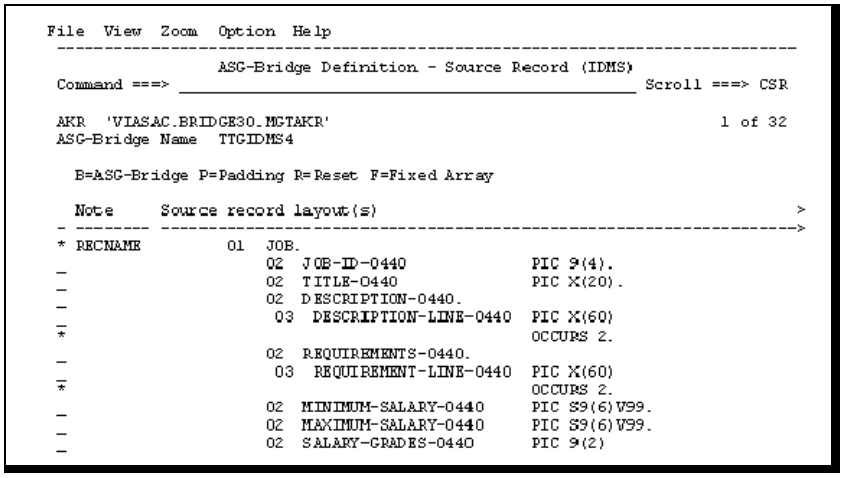


Figure 55 • ASG-Bridge Definition - Source Record Screen (IDMS)



Source Load Errors

These pop-ups display if Bridge has difficulty identifying the language of the source record you are loading or finds any syntax errors in the source record. To continue, select one of these options and press Enter:

Problem	Pop-up	Options
Cannot identify language	Language Specification (see Figure 56)	Cancel processing or specify COBOL or PL/I
Syntax errors	Source Processing Error (see Figure 57 on page 54)	Cancel processing, process as different language, or edit source

Figure 56 • Language Specification Pop-up

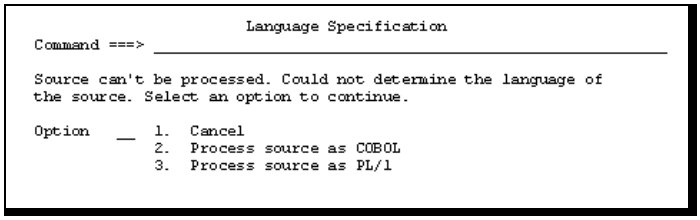
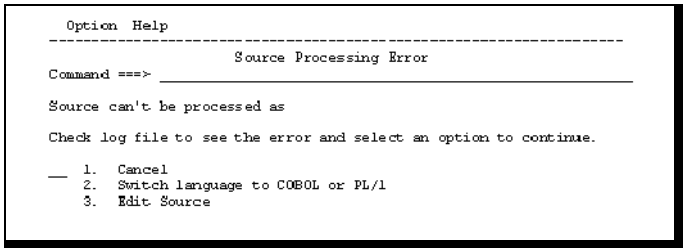


Figure 57 • Source Processing Error Pop-up



Defining Source Record Information

You must define all fields and record formats before you can convert them. The ASG-Bridge Definition - Source Record screen displays a view of the source record and provides you with line command actions and pull-downs you can use to define Bridge Fields, Bridge Pad Fields, record keys, arrays containing Bridge Fields, Test Fields, and Record Definitions. The actions available on this screen depend on the environment and Bridge Definition environment attributes that you specify. These are the environments:

- Sequential/VSAM
- CICS
- IMS/GSAM or IMS non-GSAM
- IDMS

This table describes the line command actions on the ASG-Bridge Definition - Source Record screen and what actions are available in each environment:

Action/ Command	Description	Environment
B/Bridge	Defines selected fields as Bridge Fields	All
T/Test Field	Designates selected fields as Test Fields Test Fields are assigned Field Tests to identify record formats.	Sequential/VSAM CICS IMS/GSAM
P/Padding	Defines selected fields as Pad Fields	All
R/Reset	Removes definition or designation applied to selected source lines	All
K/Key	Defines selected field as primary record key	Sequential/VSAM CICS

Action/ Command	Description	Environment
A/Alt-Key	Defines selected fields as alternate record keys	Sequential/VSAM CICS
F/Fixed Array	Designates fixed array control dataname for selected fixed array	All
S/SEG Assoc	Defines segment name for selected 01 level record to associate with record name	IMS/Non-GSAM

You can use the primary FIND or RFIND command to search for and to select items on the ASG-Bridge Definition - Source Record screen. Use the FIRST, LAST, NEXT, PREVIOUS, ALL, or APPEND operands with the FIND command to refine your search.

Use the primary command RESET to unselect items you used the FIND command to select.

Note:

You can also use the primary FIND command on all list screens and pop-ups. See "Common Commands," on page 243 for a list of commands common to all Bridge screens.

Saving or Canceling Source Record Information

To automatically save definition or designation tasks performed from the ASG-Bridge Definition - Source Record screen, follow this step:

- Press PF3 to exit the screen.

You can also save any work performed without exiting the screen.

Bridge has a cancel command you can use to cancel all definition or designation tasks performed since you last saved work done from the ASG-Bridge Definition - Source Record screen.

To save work without exiting the ASG-Bridge Definition - Source Record screen, follow this step:

- Select File ▸ Save and press Enter. Work is saved and you can continue performing definition and designation tasks.

  Or

  Type `SAVE` on the command line and press Enter.

To cancel work performed since the last save, follow this step:

- Select File ▸ Cancel and press Enter. Tasks performed since the last save are canceled. The View - ASG-Bridge Definition Directory screen displays with a message stating that changes to the specified Bridge Definition are canceled.

  Or

  Type `CANCEL` on the command line and press Enter.

Edit Source

You can directly edit the source on an ISPF edit screen and reload the edited source into Bridge. If you define any Bridge Definition elements in the source prior to an edit and confirm a reload of the edited source, you can choose to resynchronize any previous source record definitions.

To edit the source record and reload the edited source

1. Select View ▸ ASG-Bridge Definition Directory from the Bridge Primary screen and press Enter. The ASG-Bridge Definition Directory screen displays.

2. Type `V` to select a Bridge Definition and press Enter. The ASG-Bridge Definition - Source Record screen displays.

3. Select File ▸ Edit Source and press Enter. The original source displays on an ISPF edit screen.

4. Edit the source and press PF3. The Confirm Reload - Source Record pop-up displays (see Figure 58).

Figure 58 • Confirm Reload - Source Record Pop-up

```
                    Confirm Reload - Source Record
Command ===> ____________________________________________________________

Press ENTER key to confirm reload request. Source will be resynched
if resynch option is selected

  Resynch Source  _

Enter END command to cancel reload request.
```

5. Select Resynch Source to resynchronize any previous source record definitions.

6 Press Enter to reload the source and return to the ASG-Bridge Definition - Source Record screen.

Or

Press PF3 to cancel the source reload request. Canceling discards edits and returns to the ASG-Bridge Definition - Source Record screen.

Source Reload Errors from Bridge Definition - Source Record screen

If Bridge is unable to identify the language of the source record you are reloading, finds syntax errors in the source record, or cannot resynchronize source record definitions, these pop-ups display. When this happens, select one of these options and press Enter to continue:

Problem	Pop-up	Options
Cannot identify language	Language Specification (see Figure 59)	Cancel processing or specify COBOL or PL/I
Syntax errors	Source Processing Error (see Figure 60)	Cancel processing, process as different language, or edit the source record
Cannot resynchronize	Re-synchronizing Source Failed (see Figure 61 on page 58)	Cancel processing, accept partial resynchronization, or edit the source record

Figure 59 • Language Specification Pop-up

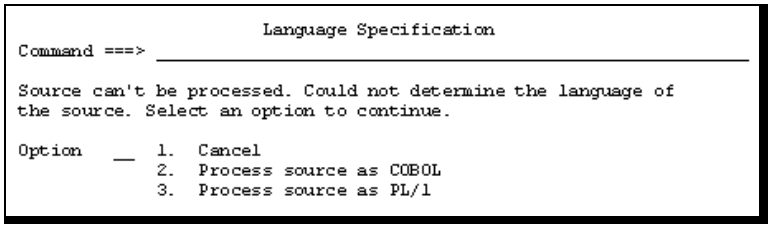


Figure 60 • Source Processing Error Pop-up (Bridge Definition - Source Record Screen)

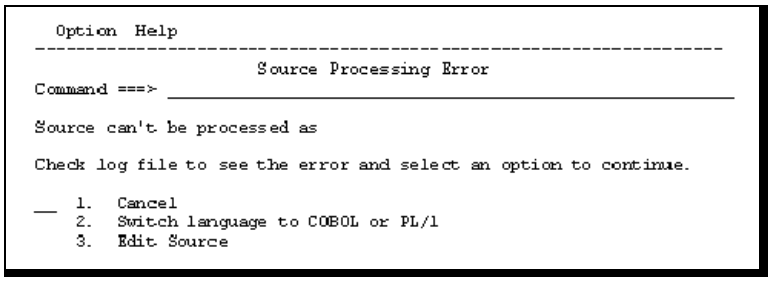
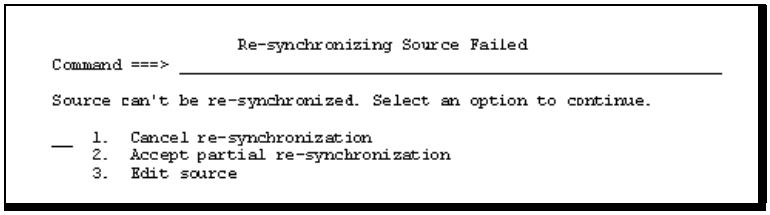


Figure 61 • Re-synchronizing Source Failed Pop-up (Bridge Definition-Source Record Screen)



Defining Bridge Fields

You associate a Bridge Field with a Bridge Rule. A Bridge Rule is a coded routine that converts fields from one format to another. Bridge provides you with Bridge Rules that include date format conversion rules and several euro conversion rules. See "Default Bridge Parameters and Rules," on page 245 for more information on the Bridge Rules that are delivered with Bridge.

To define a Bridge Field

1 Access the ASG-Bridge Definition - Source Record screen using one of these methods:

If you are building a new Bridge Definition, the ASG-Bridge Definition - Source Record screen automatically displays as described in "To name a new Bridge Definition and specify definition attributes" on page 43.

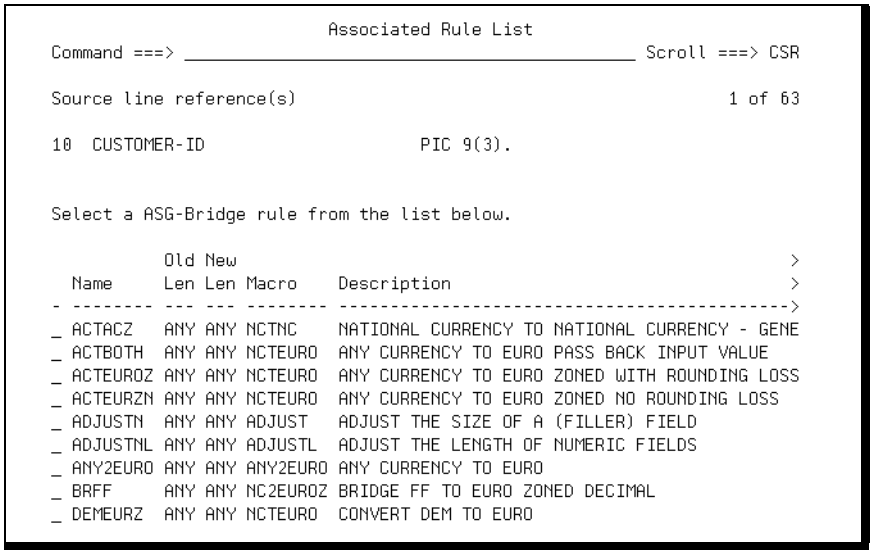
Or

Select View ▸ ASG-Bridge Definition Directory from the Bridge Primary screen and press Enter. The ASG-Bridge Definition Directory screen displays.

2 Type `V` to select a Bridge Definition and press Enter. The ASG-Bridge Definition - Source Record screen displays.

3 Type `B` to select a field to define as a Bridge Field and press Enter. The Associated Rule List pop-up displays (see Figure 62 on page 59).

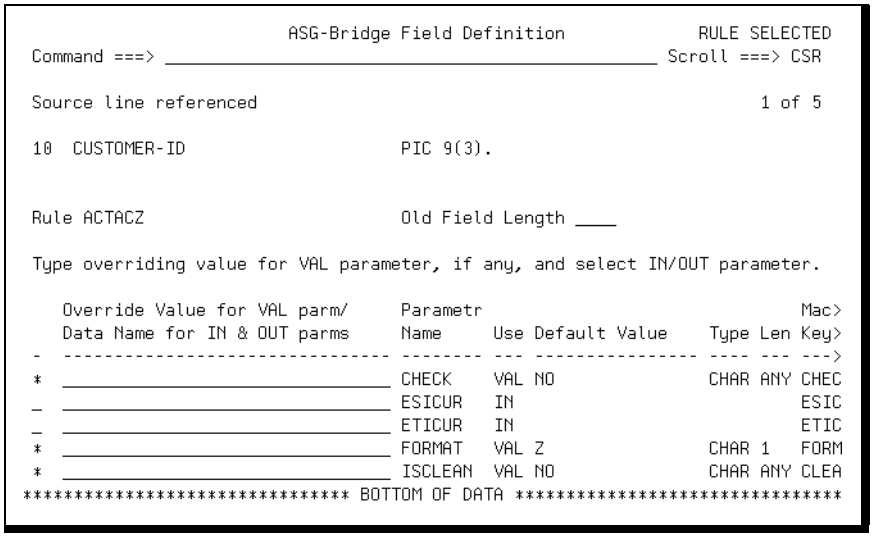
Figure 62 • Associated Rule List Pop-up



```
                          Associated Rule List
Command ===> _____________________________________________ Scroll ===> CSR

Source line reference(s)                                          1 of 63

10  CUSTOMER-ID                    PIC 9(3).


Select a ASG-Bridge rule from the list below.

           Old New                                                       >
  Name     Len Len Macro    Description                                  >
- -------- --- --- -------- ---------------------------------------------->
_ ACTACZ   ANY ANY NCTNC    NATIONAL CURRENCY TO NATIONAL CURRENCY - GENE
_ ACTBOTH  ANY ANY NCTEURO  ANY CURRENCY TO EURO PASS BACK INPUT VALUE
_ ACTEUROZ ANY ANY NCTEURO  ANY CURRENCY TO EURO ZONED WITH ROUNDING LOSS
_ ACTEURZN ANY ANY NCTEURO  ANY CURRENCY TO EURO ZONED NO ROUNDING LOSS
_ ADJUSTN  ANY ANY ADJUST   ADJUST THE SIZE OF A (FILLER) FIELD
_ ADJUSTNL ANY ANY ADJUSTL  ADJUST THE LENGTH OF NUMERIC FIELDS
_ ANY2EURO ANY ANY ANY2EURO ANY CURRENCY TO EURO
_ BRFF     ANY ANY NC2EUROZ BRIDGE FF TO EURO ZONED DECIMAL
_ DEMEURZ  ANY ANY NCTEURO  CONVERT DEM TO EURO
```

4 Select the Bridge Rule that specifies the field format you want to use and press Enter. The ASG-Bridge Field Definition pop-up displays (see Figure 63).

Figure 63 • ASG-Bridge Field Definition Pop-up



```
                      ASG-Bridge Field Definition            RULE SELECTED
Command ===> _____________________________________________ Scroll ===> CSR

Source line referenced                                            1 of 5

10  CUSTOMER-ID                  PIC 9(3).


Rule ACTACZ                      Old Field Length ____

Type overriding value for VAL parameter, if any, and select IN/OUT parameter.

   Override Value for VAL parm/     Parametr                              Mac>
   Data Name for IN & OUT parms     Name     Use Default Value    Type Len Key>
-  -------------------------------- -------- --- ---------------- ---- --- --->
*  _______________________________  CHECK    VAL NO               CHAR ANY CHEC
_  _______________________________  ESICUR   IN                            ESIC
_  _______________________________  ETICUR   IN                            ETIC
*  _______________________________  FORMAT   VAL Z                CHAR 1   FORM
*  _______________________________  ISCLEAN  VAL NO               CHAR ANY CLEA
******************************** BOTTOM OF DATA ********************************
```

5 Specify any override parameter values and select any necessary input or output fields for those parameters that are either type IN or OUT. Press PF3 to return to the ASG-Bridge Definition - Source Record screen. The Bridge Rule you defined for this field displays in the Note column.

Note:

After a Bridge Rule is defined for a field, if you use the B action to select that field on the ASG-Bridge Definition - Source Record screen to redefine the Bridge Rule, the ASG-Bridge Field Definition screen displays. To access the Associated Rule List pop-up, select Associate ▸ Rule List and press Enter.

Defining Bridge Pad Fields

A Bridge Pad Field is a field that is inserted into the new record structure to either force the alignment of subsequent fields or to adjust the length of a record. A padding field is present only in the new record structure and is defined to instruct Bridge to insert the field when bridging the record.

Padding fields are commonly used to adjust a given record to a certain length. For example, a fixed record length file may contain two types of records, each having the same length. When one record is made longer by expanding fields, adding a padding field to the shorter record is necessary to give it the same length as the other record.

Bridge automatically inserts the padding field and shifts trailing fields to the right when it bridges the record. In reverse bridging, Bridge removes the padding field and shifts the trailing fields to the left. You can designate these fill characters when you define padding fields: None, Spaces, Low Values, or High Values.

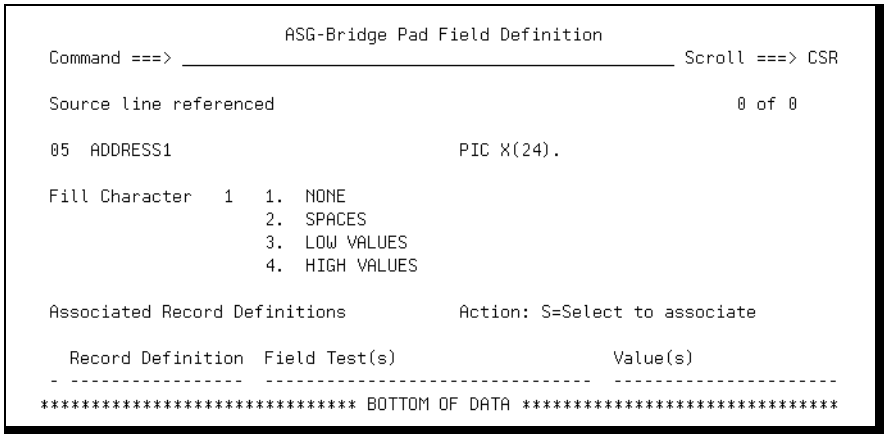
To define a Bridge Pad Field

1. Select View ▸ ASG-Bridge Definition Directory from the Bridge Primary screen and press Enter. The ASG-Bridge Definition Directory screen displays.

2. Type `V` to select a Bridge Definition and press Enter. The ASG-Bridge Definition - Source Record screen displays.

3. Type `P` to select the field you want to define as a pad field and press Enter. The ASG-Bridge Pad Field Definition screen displays (see Figure 64 on page 61).

Figure 64 • ASG-Bridge Pad Field Definition Screen



4 Designate the fill character you want to use and press PF3 to return to the ASG-Bridge Definition - Source Record screen. The field is designated as a pad field in the Note column.

Defining Primary and Alternate Record Keys

You need to define the record description fields that are used as either VSAM primary or alternate record keys to Bridge. You can define one field as a primary record key and one or more fields as alternate record keys. You cannot define an alternate record key without also defining a primary record key.

To define a Primary Record Key

To select a field to define as the primary record key, follow this step:

- Type `K` on the ASG-Bridge Definition - Source Record screen and press Enter. The Note column indicates that the field is defined as a primary record key.

To define an Alternate Record Key

To select a field to define as an alternate record key, follow this step:

- Type `A` on the ASG-Bridge Definition - Source Record screen and press Enter. The Note column indicates that the field is defined as an alternate record key.

Defining Fixed Array Control Data Name

A record may contain an array with a fixed number of occurrences. However, use of these occurrences in a record may depend upon another field in the record. This array is a hybrid form of a fixed array where some occurrences may not contain meaningful data in a record. A numeric field in the record tells the program how many occurrences are in use.

If the array contains one or more fields that require bridging, you must define this pseudo-depending on field to Bridge with a Fixed Array Control Data Name. Bridge processes the elements in the fixed array from left to right according to the value contained in the Fixed Array Control Data Name. If none of the fields contained in the array require bridging, you do not need to specify a Fixed Array Control Data Name.

You must define the Fixed Array Control Data Name in the current record structure before the array. The dataname must be a binary halfword. An error message displays if you specify an invalid dataname.

To specify a Fixed Array Control Data Name

1. Select View ▸ ASG-Bridge Definition Directory from the Bridge Primary screen and press Enter. The ASG-Bridge Definition Directory screen displays.

2. Type `V` to select a Bridge Definition and press Enter. The ASG-Bridge Definition - Source Record screen displays.

3. Type `F` to select the source line you want to specify as a Fixed Array Control Data Name and press Enter. The Generate - Fixed Array Control Data Name pop-up (see Figure 65) displays.

Figure 65 • Generate - Fixed Array Control Data Name Pop-up

```
                Generate - Fixed Array Control Data Name
Command ===> ____________________________________________________________


Source line reference(s)

05 TEST-OCCUR-VARIABLE OCCURS 9 TIMES.


Enter the name of the field which contains the number of occurrences
actually used in the specified fixed array. The field must be defined
in the current record structure before the array and must be a binary
halfword.
  i.e. for COBOL:   "S9(4) COMP"
           PL/I :   "Fixed binary (15)"

Control Data Name    TEST-OCCUR-VARIABLE-COUNT
```

4 Type a dataname in the Control Data Name field and press Enter.

Or

Select File ▸ Data Name List and press Enter. The Fixed Array - Data Name List pop-up displays a list of datanames in the current record structure (see Figure 66).

Figure 66 • Fixed Array - Data Name List Pop-up

```
                        Fixed Array - Data Name List
Command ===> _____________________________________________ Scroll ===> CSR

Select a Data Name then press Enter to continue.                 1 of 5

  Data Name in the Current Record Structure
- ------------------------------------------------------------
_ RECORD-STRUCTURE
_ TEST-OCCUR-VARIABLE-COUNT
_ TEST-VALUE-A-1
_ DATE-MMYY
_ TEST-VALUE-A-2
********* BOTTOM OF DATA *********
```

5 Select a dataname from the list and press Enter to return to the Generate - Fixed Array Control Data Name pop-up. The selected dataname displays in the Control Data Name field.

6 Select File ▸ Exit from the Generate - Fixed Array Control Data Name screen and press Enter

Or

Press PF3 to save and return to the ASG-Bridge Definition - Source Record screen. A fixed array indication displays in the Note column (see Figure 67).

Figure 67 • ASG-Bridge Definition - Source Record Screen (Showing Fixed Array Definition)

```
            ASG-Bridge Definition - Source Record (SEQ/VSAM, CICS)
Command ===> _____________________________________________ Scroll ===> CSR

AKR  'VIASAC.BRIDGE31.MGTAKR'                                    1 of 7
ASG-Bridge Name  TTGFARRY

  B=Bridge T=Test P=Padding R=Reset K=Key  A=Alt-Key  F=Fixed Array

  Note     Source record layout(s)                                      >
- -------- ------------------------------------------------------------->
_           01  RECORD-STRUCTURE.
_               05 TEST-OCCUR-VARIABLE-COUNT    PICTURE S9(4) COMP.
_ FIXARRAY      05 TEST-OCCUR-VARIABLE OCCURS 9 TIMES.
_                  10 TEST-VALUE-A-1            PICTURE X.
_ MMYY             10 DATE-MMYY                 PICTURE X(6).
_                  10 TEST-VALUE-A-2            PICTURE X.
*          *
******************************** BOTTOM OF DATA ********************************
```

Defining IMS Segment/Record Association

The IMS non-GSAM environment requires you to associate the segment name with the record name.

To define a segment/record association

1. Select View ▸ ASG-Bridge Definition Directory from the Bridge Primary screen and press Enter. The ASG-Bridge Definition Directory screen displays.

2. Type `V` to select a Bridge Definition and press Enter. The ASG-Bridge Definition - Source Record screen displays.

3. Type `S` to select the record name you want to associate with a segment name and press Enter. The Segment / Record Association pop-up displays (see Figure 68).

Figure 68 • Segment/Record Association Pop-up

```
                    Segment / Record Association
Command ===> ____________________________________________________________

Please enter segment name to associate with the record name.

Source line reference

01  SESTUDEN.

Segment Name    SESTUDEN
```

4. Type the segment name in the Segment Name field and press PF3 to save the association and to return to the Bridge Definition - Source Record screen. The segment/record association indication displays in the Note column.

Designating Test Fields and Defining Field Tests

If a file consists of multiple record formats, you must identify each record to Bridge. Records are identified by Record Definitions consisting of one or more Field Tests. If a Record Definition contains multiple Field Tests, the Field Tests are performed in the order specified by a Field Test hierarchy level. A Field Test is assigned a name and one or more test values.

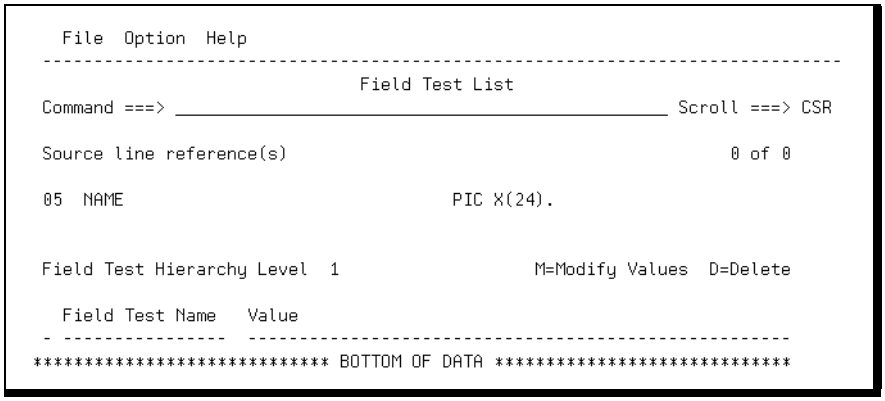
To designate a Test Field and define Field Tests

1. Select View ▸ ASG-Bridge Definition Directory from the Bridge Primary screen and press Enter. The ASG-Bridge Definition Directory screen displays.

2. Type `V` to select a Bridge Definition and press Enter. The ASG-Bridge Definition - Source Record screen displays.

3 Type T to select the field you want to define as a Test Field and press Enter. The Field Test List pop-up displays (see Figure 69).

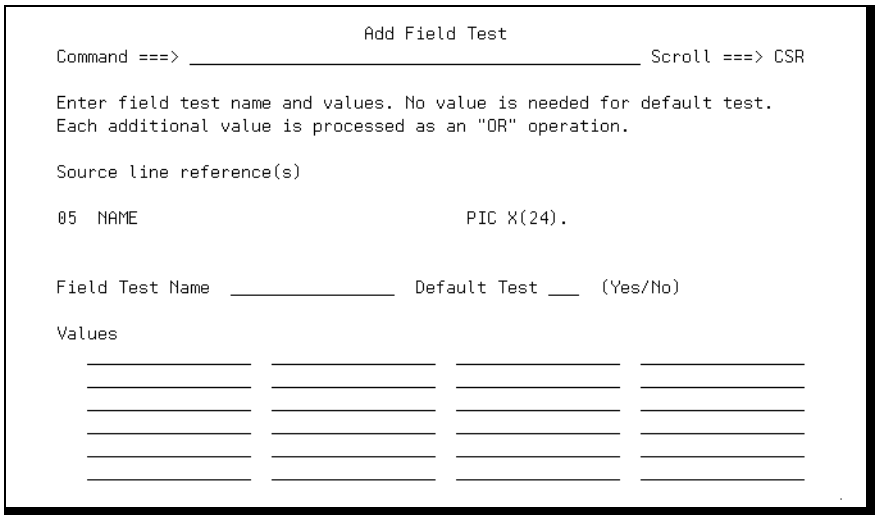
Figure 69 • Field Test List Pop-up



4 Note the Field Test Hierarchy Level is automatically assigned. You can modify this value if you want to change the order of Field Test execution.

5 Select File ▸ Add Field Test and press Enter. The Add Field Test pop-up displays (see Figure 70).

Figure 70 • Add Field Test Pop-up



6 Type a name in the Field Test Name field.

7 Indicate Yes or No in the Default Test field.

a Type `YES` to accept all values not explicitly defined in other Field Tests. No value is needed for a default test.

b Type `NO` to accept only values specified in Value fields. An abend occurs during bridging if No is specified and the Bridge Routine encounters an undefined value.

8 Type test values (up to 16 characters in length) in the Values fields. You can specify up to 100 individual values. However, you cannot specify multiple values or a range of values in an individual value field.

9 Press PF3 to save the Field Test and return to the Field Test List screen where you can continue to add, modify, or delete Field Tests.

10 Press PF3 to return to the ASG-Bridge Definition - Source Record screen when you finish defining Field Tests for the field. An indication displays in the Note column designating the field as a Test Field and shows the Field Test hierarchy level (see Figure 71).

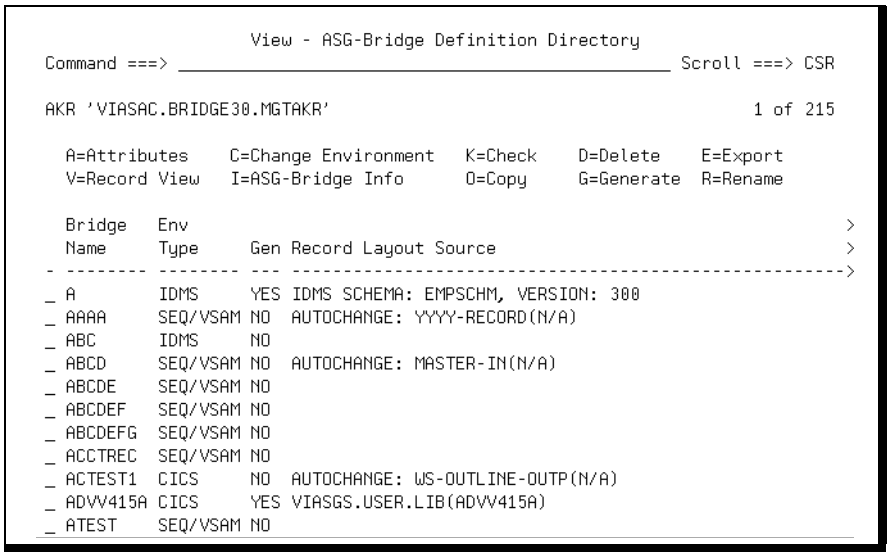
Figure 71 • ASG-Bridge Definition - Source Record Screen Test Field Designation

```
  File  View  Zoom  Option  Help
--------------------------------------------------------------------------------
           ASG-Bridge Definition - Source Record (SEQ/VSAM, CICS)
Command ===> _____________________________________________ Scroll ===> CSR

AKR  'VIASAC.BRIDGE31.MGTAKR'                                        1 of 24
ASG-Bridge Name  ACTST1

  B=Bridge T=Test P=Padding R=Reset K=Key  A=Alt-Key  F=Fixed Array

  Note     Source record layout(s)                                          >
- -------- ------------------------------------------------------------------>
_             01  MASTER-IN.
_                 05 CLIENT-ID.
_ PRIM KEY            10  DISTRICT-ID                  PIC 9(3).
_ ACTACZ              10  CUSTOMER-ID                  PIC 9(3).
_ TSTFLD 1        05  NAME                             PIC X(24).
_ PAD             05  ADDRESS1                         PIC X(24).
_                 05  CITY                             PIC X(20).
_                 05  STATE                            PIC X(2).
_                 05  ZIP.
_                     10  ZIP-CODE                     PIC 9(5).
_                     10  FILLER                       PIC 9(11).
_                 05  PHONE.
```

To modify a Field Test Definition

1 Select View ▸ ASG-Bridge Definition Directory from the Bridge Primary screen and press Enter. The ASG-Bridge Definition Directory screen displays (see Figure 72).

Figure 72 • ASG-Bridge Definition - Source Record Screen



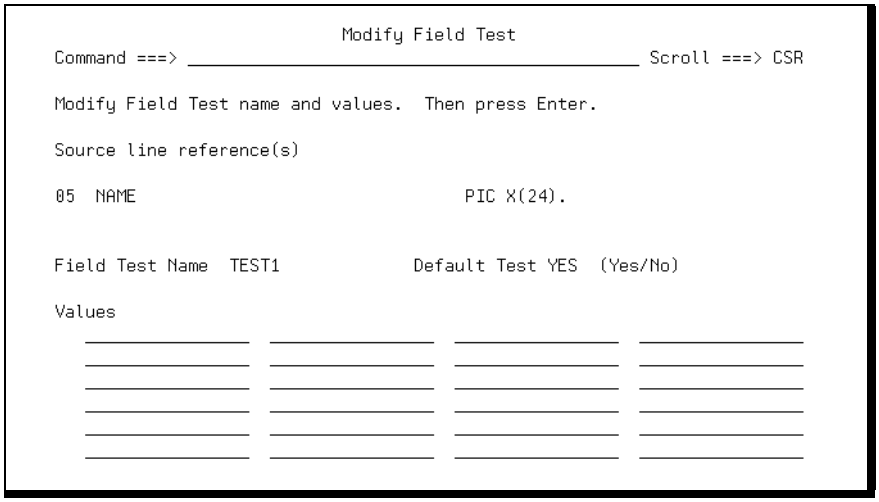
```
                    View - ASG-Bridge Definition Directory
Command ===> _____________________________________________ Scroll ===> CSR

AKR 'VIASAC.BRIDGE30.MGTAKR'                                      1 of 215

  A=Attributes    C=Change Environment   K=Check    D=Delete    E=Export
  V=Record View   I=ASG-Bridge Info      O=Copy     G=Generate  R=Rename

  Bridge   Env                                                          >
  Name     Type     Gen Record Layout Source                            >
- -------- -------- --- ------------------------------------------------->
_ A        IDMS     YES IDMS SCHEMA: EMPSCHM, VERSION: 300
_ AAAA     SEQ/VSAM NO  AUTOCHANGE: YYYY-RECORD(N/A)
_ ABC      IDMS     NO
_ ABCD     SEQ/VSAM NO  AUTOCHANGE: MASTER-IN(N/A)
_ ABCDE    SEQ/VSAM NO
_ ABCDEF   SEQ/VSAM NO
_ ABCDEFG  SEQ/VSAM NO
_ ACCTREC  SEQ/VSAM NO
_ ACTEST1  CICS     NO  AUTOCHANGE: WS-OUTLINE-OUTP(N/A)
_ ADVV415A CICS     YES VIASGS.USER.LIB(ADVV415A)
_ ATEST    SEQ/VSAM NO
```

2 Type `V` to select a Bridge Definition and press Enter. The ASG-Bridge Definition - Source Record screen displays.

3 Type `T` to select the Test Field that is assigned the Field Test you want to modify and press Enter. The Field Test List pop-up displays.

4 Type `M` to select the Field Test you want to modify and press Enter. The Modify Field Test pop-up displays (see Figure 73 on page 68).

Figure 73 • Modify Field Test Pop-up



5 Make modifications and press PF3 to save the changes and return to the Field Test List pop-up.

6 Press PF3 to return to the ASG-Bridge Definition - Source Record screen.

To delete a Field Test Definition

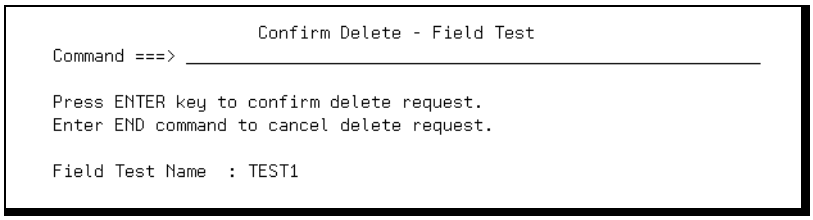
1 Select View ▸ ASG-Bridge Definition Directory from the Bridge Primary screen and press Enter. The ASG-Bridge Definition Directory screen displays.

2 Type `V` to select a Bridge Definition and press Enter. The ASG-Bridge Definition - Source Record screen displays.

3 Type `T` to select the Test Field that is assigned the Field test you want to delete and press Enter. The Field Test List pop-up displays.

4 Type `D` to select the Field Test you want to delete and press Enter. The Confirm Delete - Field Test pop-up displays (see Figure 74).

Figure 74 • Confirm Delete - Field Test Pop-up



5 Press Enter to confirm.

Or

Type `END` to cancel the deletion and return to the Field Test List pop-up.

6 Press PF3 to return to the ASG-Bridge Definition - Source Record screen.

Creating Record Definitions

Bridge identifies record formats by Record Definitions consisting of one or more Field Tests. The Record Definition identifies the record layout name and is then associated with one or more Bridge Fields in the record.

To create a Record Definition

1 Select View ▸ ASG-Bridge Definition Directory from the Bridge Primary screen and press Enter. The ASG-Bridge Definition Directory screen displays.

2 Type `V` to select a Bridge Definition and press Enter. The ASG-Bridge Definition - Source Record screen displays.

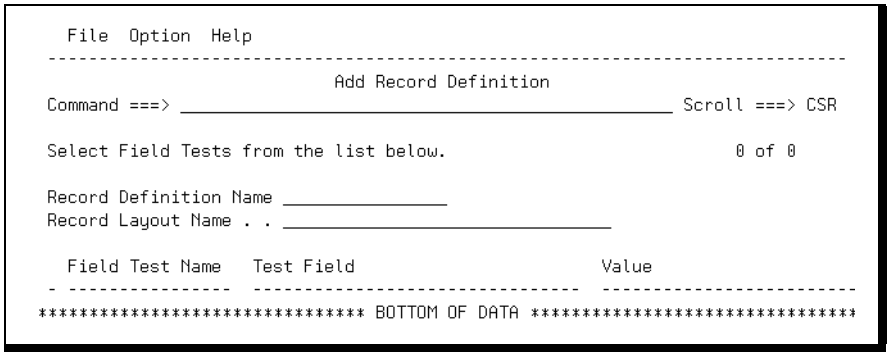
3 Select File ▸ Record Definition and press Enter. The Record Definition List pop-up displays (see Figure 75).

Figure 75 • Record Definition List Pop-up

```
  File  Associate  Option  Help
--------------------------------------------------------------------------------
                        Record Definition List
Command ===> _____________________________________________ Scroll ===> CSR

M=Modify  B=ASG-Bridge Associations D=Delete                        0 of 0

  Record Definition  Record Layout Name               Field Test Names
- -----------------  -------------------------------  ------------------
****************************** BOTTOM OF DATA ******************************
```

4 Select File ▸ Add Record Definition and press Enter. The Add Record Definition screen displays (see Figure 76 on page 70).

Figure 76 • Add Record Definition Screen



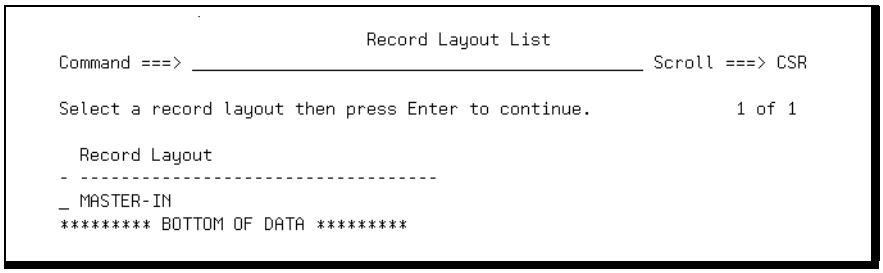
5 Type a name in the Record Definition Name field.

6 Type a record layout name exactly as it displays on the ASG-Bridge Definition - Source Record screen in the Record Layout Name field.

Or

Select File ▸ Select Record Layout Name and press Enter. The Record Layout List pop-up displays listing the available record layouts (see Figure 77).

Figure 77 • Record Layout List Pop-up



7 Select a record layout and press PF3 to return to the Add Record Definition screen. The selected record layout name displays in the Record Layout Name field.

8 Select the Field Tests you want to assign to this Record Definition.

9 Press PF3 to add the Record Definition and to return to the Record Definition List screen where the Record Definition and assigned Field Test Names are listed.

10 Press PF3 to return to the ASG-Bridge Definition - Source Record screen.

To modify a Record Definition

1 Select View ▸ ASG-Bridge Definition Directory from the Bridge Primary screen and press Enter. The ASG-Bridge Definition Directory screen displays.

2 Type V to select a Bridge Definition and press Enter. The ASG-Bridge Definition - Source Record screen displays.

3 Select File ▸ Record Definition and press Enter. The Record Definition List pop-up displays.

4 Type M to select a Record Definition to modify and press Enter. The Modify Record Definition screen displays (see Figure 78).

Figure 78 • Modify Record Definition Screen

```
  File  Option  Help
--------------------------------------------------------------------------------
                          Modify Record Definition
Command ===> _____________________________________________ Scroll ===> CSR

Select Field Tests from the list below.                          1 of 2

Record Definition Name TESTS
Record Layout Name . . MASTER-IN

  Field Test Name   Test Field                      Value
- ---------------  --------------------------------  -------------------------
/ TEST2            CITY                           *DEFAULT TEST*
_ TEST3            CITY                           23
****************************** BOTTOM OF DATA ********************************
```

5 Make the modifications and press PF3 to save and return to the Record Definition List pop-up.

6 If you change the Record Layout Name of a Record Definition that is associated with Bridge Fields, the associations become invalid. The Confirm Bridge List Deassociation pop-up displays to verify that you want to continue with the change (see Figure 79).

Figure 79 • Confirm Bridge List Deassociation Pop-up

```
                Confirm ASG-Bridge List Deassociation
Command ===> _____________________________________ Scroll ===> CSR

Changing Record Layout Name makes ASG-Bridge field associations invalid.
Press ENTER key to confirm deassociation.
Enter END command to cancel deassociation.

Record Definition Name RECORD1                          1 of 1

ASG-Bridge Field                Rule
------------------------------  --------
DATE-4                          DDMMYY
************* BOTTOM OF DATA *************
```

7 Press Enter to confirm the change and return to the Record Definition List pop-up. A message displays verifying the deassociation.

Or

Press PF3 to cancel the change and return to the Modify Record Definition pop-up. A message displays verifying the deassociation is canceled. Press PF3 to return to the Record Definition List pop-up.

8 Press PF3 to return to the ASG-Bridge Definition - Source Record screen.

To delete a Record Definition

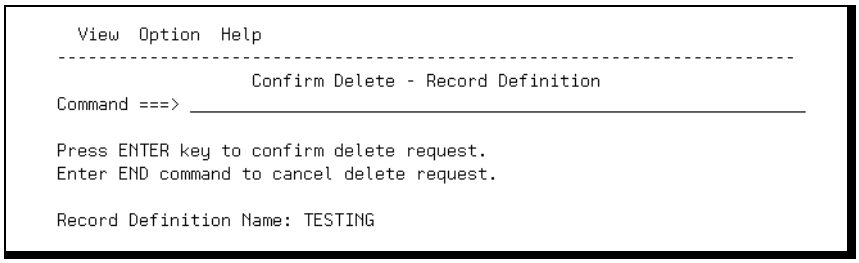
1 Select View ▸ ASG-Bridge Definition Directory from the Bridge Primary screen and press Enter. The ASG-Bridge Definition Directory screen displays.

2 Type `V` to select a Bridge Definition and press Enter. The ASG-Bridge Definition - Source Record screen displays.

3 Select File ▸ Record Definition and press Enter. The Record Definition List pop-up displays.

4 Type `D` to select a Record Definition to delete and press Enter. The Confirm Delete - Record Definition pop-up displays (see Figure 80).

Figure 80 • Confirm Delete - Record Definition Pop-up



5 Press Enter to confirm.

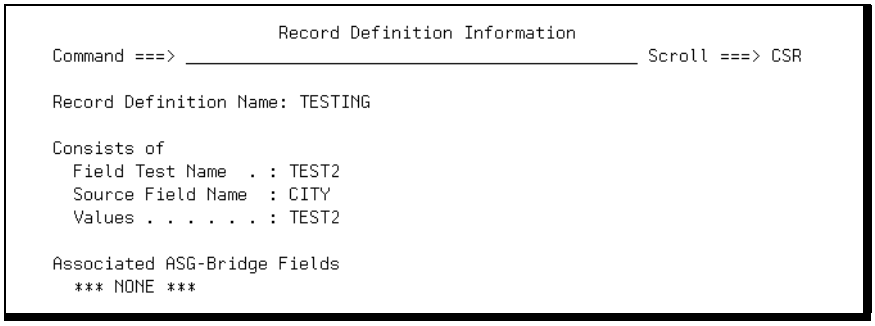
Or

Type `END` to cancel the deletion and return to the Record Definition List pop-up.

To review the Record Definition information before confirming the delete request

1 Select View ▸ Record Definition Information from the Confirm Delete - Record Definition Information pop-up and press Enter. The Record Definition Information pop-up displays (see Figure 81).

Figure 81 • Record Definition Information Pop-up



2 Press PF3 to return to the Confirm Delete - Record Definition pop-up.

Associating Bridge and Pad Fields with Record Definitions

If you defined multiple Record Definitions for your file, you must associate each Bridge and Pad field to its related Record Definitions. If Bridge detects an association error when you attempt to exit, save, or generate from the ASG-Bridge Definition - Source Record screen, a screen displays indicating the type of error.

Associations can be made by either of these methods. The only difference lies in the method of association.

- Associate the Bridge or Pad Fields to the Record Definition.
- Associate the Record Definition to the Bridge or Pad Fields.

You can delete associations between Bridge or Pad fields and Record Definitions if necessary.

To associate Pad Fields with Record Definitions

1 Select View ▸ ASG-Bridge Definition Directory from the Bridge Primary screen and press Enter. The ASG-Bridge Definition Directory screen displays.

2 Type `V` to select a Bridge Definition and press Enter. The ASG-Bridge Definition - Source Record screen displays.

3 Type `P` to select the Bridge Field to pad and press Enter. The ASG-Bridge Pad Field Definition pop-up displays.

4 Type S to select the Record Definitions to associate to the Pad Field and press PF3 to save the associations and to return to the ASG-Bridge Definition - Source Record screen.

To associate Bridge Fields with Record Definitions

1 Select Associate ▸ Record Definition from the ASG-Bridge Field Definition screen (see "To define a Bridge Field" on page 58) and press Enter. The ASG-Bridge Field Record Definition screen displays (see Figure 82).

Figure 82 • ASG-Bridge Field Record Definition Screen

```
  Zoom  Option  Help
--------------------------------------------------------------------------------
                    ASG-Bridge Field Record Definition
Command ===> _____________________________________________ Scroll ===> CSR

Source line referenced                                            1 of 2

05  STATE                              PIC X(2).


Associated Record Definitions         Action : S=Select to associate

  Record Definition  Field Test(s)                  Value(s)
- -----------------  -------------------------------  ----------------------
_ FUBA               TEST3                            23
_ TESTING            TEST2                            TEST2
****************************** BOTTOM OF DATA ******************************
```

Note:

Record Definitions associated with the specified Bridge Field are marked with a forward slash (/) when you return to this screen after the initial association.

2 Type S to select the Record Definitions to associate to the Bridge Field and press PF3 to save the associations.

3 Press PF3 to return to the ASG-Bridge Field Definition screen. Press PF3 again to return to the ASG-Bridge Definition - Source Record screen.

To associate Record Definitions to Bridge or Bridge Pad fields

1 Select View ▸ ASG-Bridge Definition Directory from the Bridge Primary screen and press Enter. The ASG-Bridge Definition Directory screen displays.

2 Type V to select a Bridge Definition. The ASG-Bridge Definition - Source Record screen displays.

3 Select File ▸ Record Definition and press Enter. The Record Definition List pop-up displays.

4 Type B to select the Record Definition to associate to Bridge Fields or Bridge Pad Fields and press Enter. The Associated ASG-Bridge Fields pop-up displays (see Figure 83).

Figure 83 • Associated ASG-Bridge Fields Pop-up

```
                    Associated ASG-Bridge Fields
Command ===> _____________________________________ Scroll ===> CSR

Select ASG-Bridge Fields from the list below.              1 of 4

Record Definition Name TESTING

  ASG-Bridge Field                 Rule
- -------------------------------  --------
/ CUSTOMER-ID                     ACTACZ
_ ADDRESS1                        PADDING
/ STATE                           AA
_ PHONE-NUMBER                    YY
************** BOTTOM OF DATA **************
```

5 Select the Bridge Fields or Bridge Pad Fields to associate with the selected Record Definition.

6 Press PF3 to return to the Record Definition List pop-up.

7 Press PF3 to return to the ASG-Bridge Definition - Source Record screen.

Note:

Bridge Fields associated with the specified Record Definition are marked with a forward slash (/) when you return to this screen after the initial association.

Association Errors

The ASG-Bridge Field/Definition Association Error screen displays if Bridge detects errors in Bridge or Pad Field associations with Record Definitions when you attempt to exit, save, or generate from the ASG-Bridge Definition - Source Record screen (see Figure 84). You must resolve the errors to proceed.

Figure 84 • ASG-Bridge Field/Definition Association Error Screen

```
File  Option  Help
 ------------------------------------------------------------------------
            ASG-Bridge Field/Definition Association Error
 Command ===> _____________________________________ Scroll ===> CSR

 ASG-Bridge Field/Record definition association errors have been
 detected.  Note the error number from the Err column below and press
 the help PF key for more information.

 Err  ASG-Bridge Field              Associated Record Definitions
 ---  ----------------------------  ------------------------------------
****************************** BOTTOM OF DATA ******************************
```

Press PF1 to access online help on this screen and to view a list explaining possible errors and suggested solutions, or see "Bridge Field and Record Definition Association Errors," on page 185.

Deleting Associations

You can delete associations between Bridge or Pad Fields and Record Definitions to resolve association errors, or to remove an association that is no longer valid.

To delete Record Definition and Bridge or Pad Field associations

1. Select View ▸ ASG-Bridge Definition Directory from the Bridge Primary screen. and press Enter The ASG-Bridge Definition Directory screen displays.

2. Type `V` to select a Bridge Definition and press Enter. The ASG-Bridge Definition - Source Record screen displays.

3. Type `P` to select the Bridge Field to pad and press Enter. The ASG-Bridge Pad Field Definition pop-up displays.

4. Unselect the associated Record Definitions to delete the associations with the selected Bridge or Pad Fields.

   Or

   Access the Record Definition List pop-up.

5. Type `B` to select the Record Definition that contains Bridge Field associations that you want to delete. The Associated ASG-Bridge Fields pop-up displays (see Figure 85).

Figure 85 • Associated ASG-Bridge Field Pop-up

```
                  Associated ASG-Bridge Fields
Command ===> _____________________________________ Scroll ===> CSR

Select ASG-Bridge Fields from the list below.              1 of 4

Record Definition Name TESTING

  ASG-Bridge Field                 Rule
- -------------------------------  --------
/ CUSTOMER-ID                     ACTACZ
_ ADDRESS1                        PADDING
/ STATE                           AA
_ PHONE-NUMBER                    YY
************** BOTTOM OF DATA **************
```

6. Unselect the Bridge Fields or Bridge Pad Fields to delete the association with the selected Record Definition.

7 Press PF3 to return to the Record Definition List pop-up.

8 Press PF3 to return to the ASG-Bridge Definition - Source Record screen.

Removing Source Record Definitions or Designations

Use to reset the action to remove the definition or designation applied to any line of the source record.

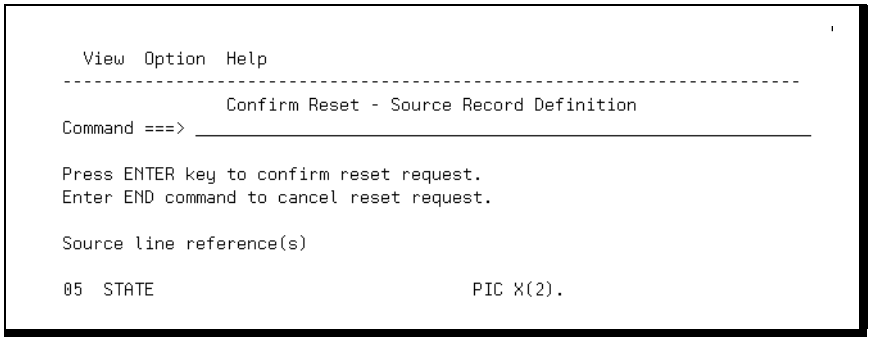
To reset a defined line on the Bridge Definition - Source Record screen

1 Select View ▸ ASG-Bridge Definition Directory from the Bridge Primary screen and press Enter. The ASG-Bridge Definition Directory screen displays.

2 Type `V` to select a Bridge Definition and press Enter. The ASG-Bridge Definition - Source Record screen displays.

3 Type `R` to select the line you want to reset and press Enter. The Confirm Reset - Source Record Definition pop-up displays (see Figure 86).

Figure 86 • Confirm Reset - Source Record Definition Pop-up



4 Press Enter to confirm.

Or

Use the END command to cancel the reset request and return to the ASG-Bridge Definition - Source Record screen.

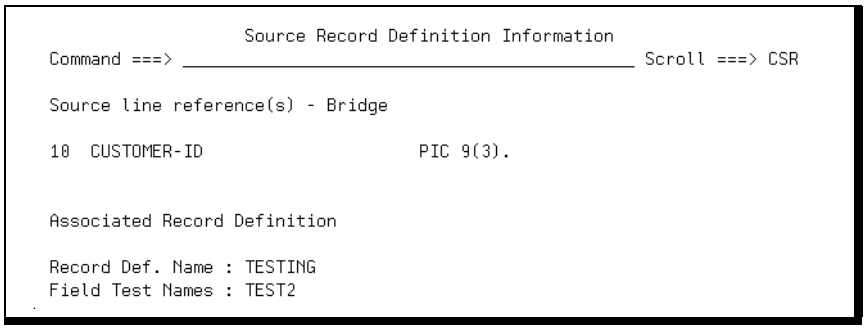
5 Press PF3 to return to the Confirm Reset - Source Record Definition pop-up.

To view source record information

1 Select View ▸ Source Record Definition from the Confirm Reset - Source Record Definition pop-up and press Enter. The Source Record Definition Information pop-up displays (see Figure 87).

2 Press PF3 to return to the Confirm Reset - Source Record definition pop-up.

Figure 87 • Source Record Definition Information Pop-up



Zoom In/Zoom Out Feature

Bridge has a zoom feature that is available on the ASG-Bridge Definition - Source Record screen, the ASG-Bridge Field Record Definition screen, and the ASG-Bridge Pad Field Definition screen. The zoom feature toggles the view on the ASG-Bridge Definition - Source Record screen to show only the source lines currently defined or all of the source lines. It toggles the views on the ASG-Bridge Field Record Definition and ASG-Bridge Pad Field Definition screens to show all the Record Definitions, or only the associated Record Definitions.

To use the Zoom feature, follow this step:

- Select Zoom ▸ Zoom in or Zoom out and press Enter.

  Or

  Type `ZOOM` on the command line and press Enter to toggle between Zoom in and Zoom out.

Viewing Bridge Definition Summary Information

You can view a statistical summary of the defined components of a Bridge Definition. The summary information lists the various components of a Bridge Definition and shows the total number of each type of component currently defined for the Bridge Definition. You can view Bridge Definition summary information from the Bridge Definition - Source Record, the View - Bridge Definition Directory, or the View - Bridge Definition Cross Reference screen.

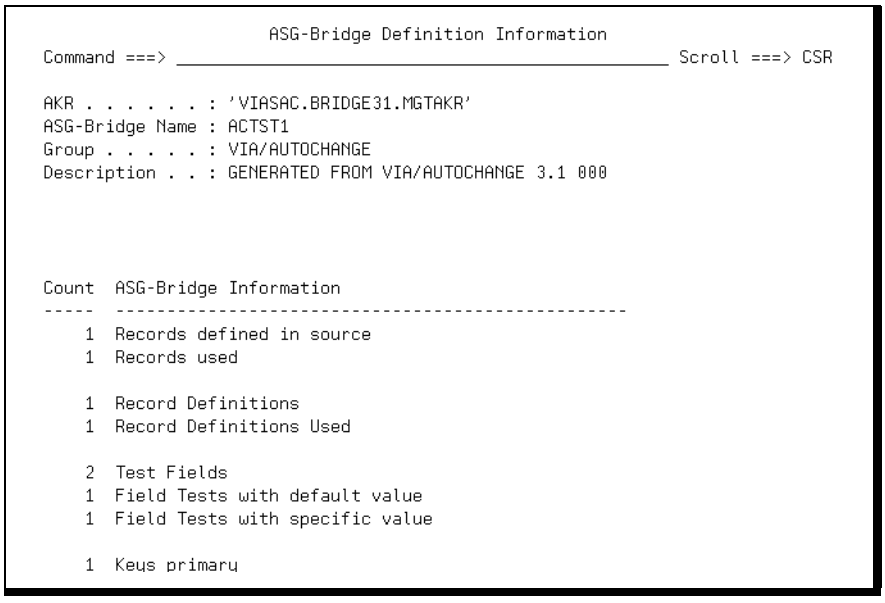
To view Bridge Definition Information from the ASG-Bridge Definition - Source Record screen

1 Select View ▸ ASG-Bridge Definition Directory from the Bridge Primary screen and press Enter. The ASG-Bridge Definition Directory screen displays.

2 Type V to select a Bridge Definition and press Enter. The ASG-Bridge Definition - Source Record screen displays.

3 Select View ▸ Bridge Definition Information and press Enter. The ASG-Bridge Definition Information screen displays (see Figure 88).

Figure 88 • ASG-Bridge Definition Information Screen



4 Review the summary information on the ASG-Bridge Definition Information screen and press PF3 to return to the ASG-Bridge Definition - Source Record screen.

To view Bridge Definition Information from the View - ASG-Bridge Definition Directory or the ASG-View - Bridge Definition Cross Reference screen

1. Access the View - ASG-Bridge Definition Directory screen (see "Bridge Definition Directory" on page 80) or the View - ASG-Bridge Definition Cross Reference screen and press Enter (see "Cross-Reference Feature" on page 109).

2. Type `I` to select a Bridge Definition to view its summary information and press Enter. The ASG-Bridge Definition Information screen displays (see Figure 89 on page 81).

3. Review the summary information. Press PF3 to return to the View - ASG-Bridge Definition Directory or View - ASG-Bridge Definition Cross Reference screen.

Bridge Definition Directory

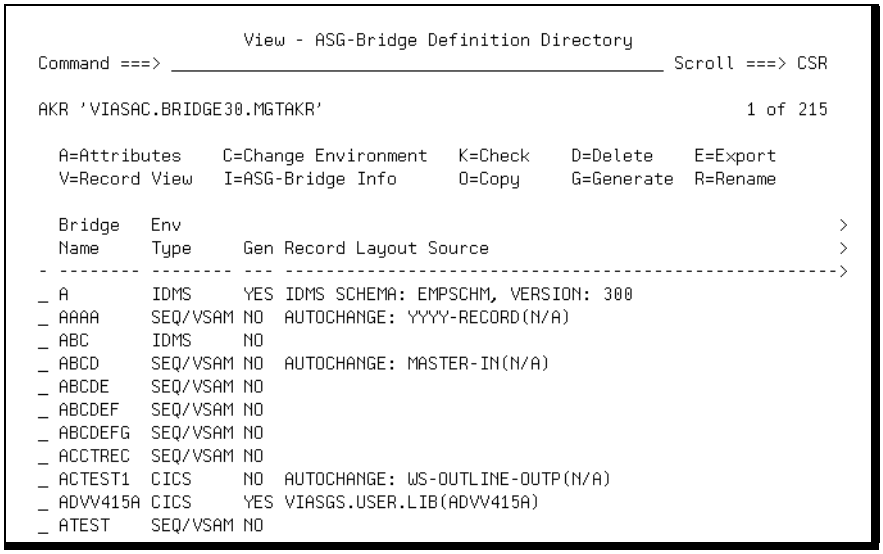
After you build and save a Bridge Definition, you can access it from the View - ASG-Bridge Definition Directory screen (see Figure 89 on page 81). This screen displays a directory of all the Bridge Definitions that exist in the specified AKR. You can use line command actions and File pull-down actions on this screen to access other Bridge screens and pop-ups.

You can use the primary FIND command to search for and select Bridge Definitions. Use the FIRST, LAST, NEXT, PREVIOUS, ALL, or APPEND operands with the FIND command to refine your search.

Note:

You can also use the primary FIND command on all list screens and pop-ups. See "Common Commands," on page 243 for a list of commands common to all Bridge screens.

Figure 89 • View - ASG-Bridge Definition Directory Screen



```
                    View - ASG-Bridge Definition Directory
Command ===> _____________________________________________ Scroll ===> CSR

AKR 'VIASAC.BRIDGE30.MGTAKR'                                      1 of 215

  A=Attributes    C=Change Environment   K=Check    D=Delete    E=Export
  V=Record View   I=ASG-Bridge Info      O=Copy     G=Generate  R=Rename

  Bridge   Env                                                          >
  Name     Type     Gen Record Layout Source                            >
- -------- -------- --- ------------------------------------------------->
_ A        IDMS     YES IDMS SCHEMA: EMPSCHM, VERSION: 300
_ AAAA     SEQ/VSAM NO  AUTOCHANGE: YYYY-RECORD(N/A)
_ ABC      IDMS     NO
_ ABCD     SEQ/VSAM NO  AUTOCHANGE: MASTER-IN(N/A)
_ ABCDE    SEQ/VSAM NO
_ ABCDEF   SEQ/VSAM NO
_ ABCDEFG  SEQ/VSAM NO
_ ACCTREC  SEQ/VSAM NO
_ ACTEST1  CICS     NO  AUTOCHANGE: WS-OUTLINE-OUTP(N/A)
_ ADVV415A CICS     YES VIASGS.USER.LIB(ADVV415A)
_ ATEST    SEQ/VSAM NO
```

This are the tasks you can access from the View - ASG-Bridge Definition Directory screen:

To perform this task…	Select…
Select AKR that contains Bridge Definitions to modify or review	File ▸ Select AKR from the File pull-down. Alternately, type an AKR name in the AKR field.
Review or modify definition attributes	Bridge Definition with the A action.
Define source information	Bridge Definition with the V action.
Change definition environment type	Bridge Definition with the C action.
View statistical definition information	Bridge Definition with the I action.
Check generate status of definition	Bridge Definition with the K action.
Copy Bridge Definition	Bridge Definition with the O action.
Delete a Bridge Definition	Bridge Definition with the D action.
Generate Bridge Routines from Bridge Definitions	Bridge Definition with the G action or select File ▸ Generate from the File pull-down. See "Generating Executable Bridge Routines" on page 95.

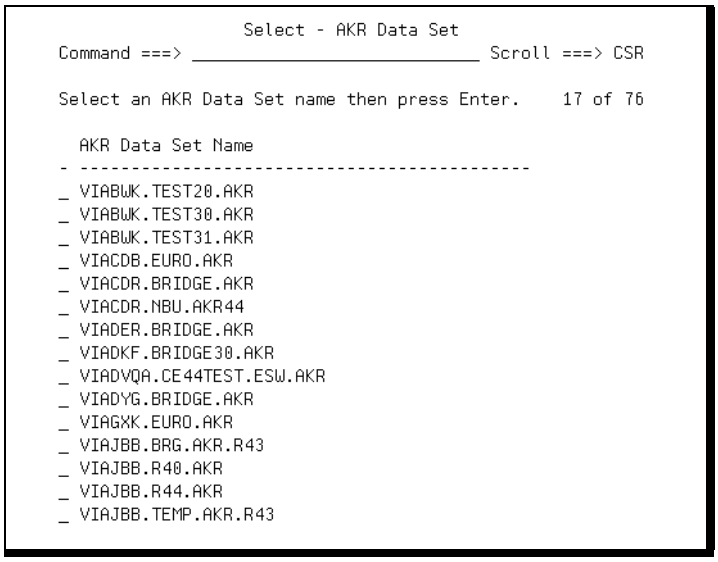
To perform this task…	Select…
Export Bridge Definitions	Bridge Definition with the E action. See "Importing/Exporting Parameters, Rules, and Definitions" on page 115.
Rename an existing definition	Bridge Definition with the R action.

To view an AKR from the View - ASG-Bridge Definition Directory screen

1 Select View ▸ ASG-Bridge Definition Directory from the Bridge Primary screen and press Enter. The View - ASG-Bridge Definition Directory screen displays.

2 Select File ▸ Select AKR and press Enter. The Select - AKR Data Set pop-up displays (see Figure 90).

Figure 90 • Select - AKR Data Set Pop-up



3 Select the AKR you want to view and press Enter to return to the View - ASG-Bridge Definition Directory screen. Bridge Definitions in the selected AKR display on the screen.

Or

Type the AKR dataset name in the AKR field on the View - ASG-Bridge Definition Directory screen and press Enter.

Copying a Bridge Definition

You can copy an existing Bridge Definition to create a new Definition with the same attributes.

To copy a Bridge Definition

1 Select View ▸ ASG-Bridge Definition Directory from the Bridge Primary screen and press Enter. The View - ASG-Bridge Definition Directory screen displays.

2 Type O to select the definition you want to copy and press Enter. The ASG-Bridge Definition copy pop-up displays (see Figure 91).

Figure 91 • ASG-Bridge Definition Copy Pop-up

```
                    ASG-Bridge Definition copy
Command ===> ____________________________________________________________

Enter a ASG-Bridge definition name to copy into, then press ENTER to
confirm.

Copy from ASG-Bridge Definition Name  : ACTST1
Copy into ASG-Bridge Definition Name    ________
```

3 Specify a name in the Copy into Bridge Definition Name field. Press Enter to save the new Definition and return to the View - ASG-Bridge Definition Directory screen.

Checking the Generate Status of a Bridge Definition

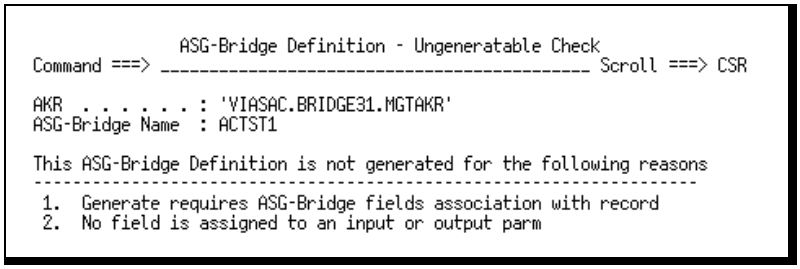
The Gen column on the View - ASG-Bridge Definition Directory screen indicates whether or not the Bridge Definition can be generated. If the status is No, you can use the Check action to determine why the Definition cannot be generated.

To check the generate status of a Bridge Definition

1 Select View ▸ ASG-Bridge Definition Directory from the Bridge Primary screen and press Enter. The View - ASG-Bridge Definition Directory screen displays.

2 Type K to select the definition with a No status that you want to check and press Enter. The ASG-Bridge Definition - Ungeneratable Check pop-up displays listing the reasons the Definition cannot be generated (see Figure 92 on page 84).

Figure 92 • ASG-Bridge Definition - Ungeneratable Check Pop-up



Modifying Existing Bridge Definitions

You can change a Bridge Definition name or environment, modify attributes, and change source Record Definitions. If you make any of these changes, you must regenerate the Bridge Routine in order to incorporate the changes.

Renaming an Existing Bridge Definition

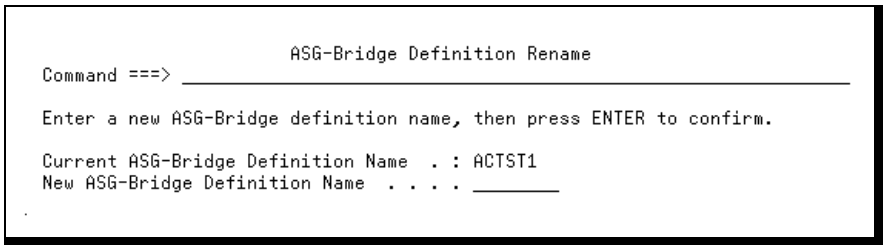
You can rename an existing Bridge Definition to reflect different information than what the original name conveys. The Bridge Definition name is informative only, and changing it does not change the name of the Bridge Routine that is generated from the original Bridge Definition. However, since the Bridge Routine name is derived from the Bridge Definition name, ASG recommends regenerating the Bridge Routine so the names of both the Bridge Definition and the Bridge Routine indicate the relationship between the two. The old Bridge Routine is not deleted, so it may be necessary to manually delete it if you are no longer using it.

To rename a Bridge Definition

1 Select View ▸ ASG-Bridge Definition Directory from the Bridge Primary screen and press Enter. The ASG-Bridge Definition Directory screen displays.

2 Type `R` to select the definition you want to rename and press Enter. The ASG-Bridge Definition Rename pop-up displays (see Figure 93).

Figure 93 • ASG-Bridge Definition Rename Pop-up



3 Type the new name in the New Bridge Definition Name field. A message displays if you attempt to use the name of an existing Bridge Definition.

4 Press Enter to confirm and save the name change and return to the View - ASG-Bridge Definition Directory screen.

Modifying Bridge Definition Group and Description

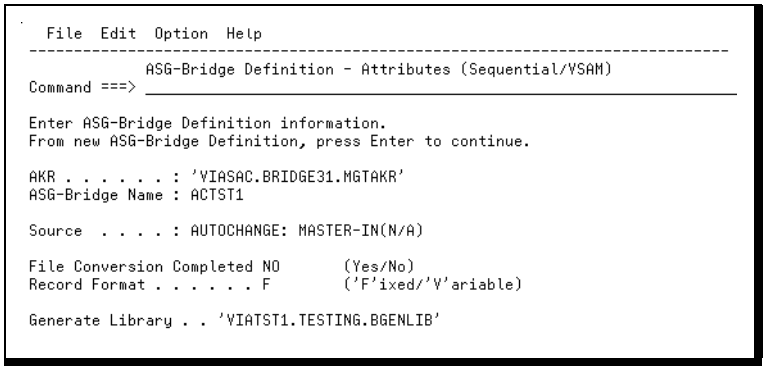
The group and description parameters are for organization and information purposes and do not effect the Bridge Routine. If you change the group name or the long description of a Bridge Definition, you do not need to regenerate the Bridge Routine.

To change a Bridge Definition Group or Description

1 Select View ▸ ASG-Bridge Definition Directory from the Bridge Primary screen and press Enter. The ASG-Bridge Definition Directory screen displays.

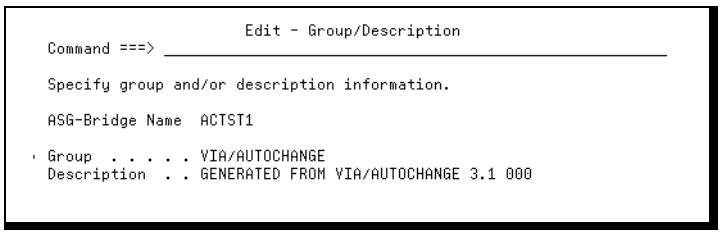
2 Type A to select the definition you want to change and press Enter. The ASG-Bridge Definition - Attributes screen displays (see Figure 94).

Figure 94 • ASG-Bridge Definition - Attributes Screen



3 Select Edit ▸ Group/Description and press Enter. The Edit - Group/Description pop-up displays (see Figure 95).

Figure 95 • Edit - Group/Description Pop-up



4 Type the new group or long description in the appropriate field, press Enter to save the changes. Press PF3 to return to the ASG-Bridge Definition - Attributes screen.

Changing a Bridge Definition Environment

You can change the environment specified for a Bridge Definition. However, depending on the environment you are changing to, some source Record Definitions or attribute information may be deleted if it is not applicable to the new environment.

If the environment you are changing to requires additional attribute information, the appropriate ASG-Bridge Definition - Attributes screen displays automatically allowing you to supply the required information.

This table shows the source Record Definitions and attribute information that are applicable to each environment:

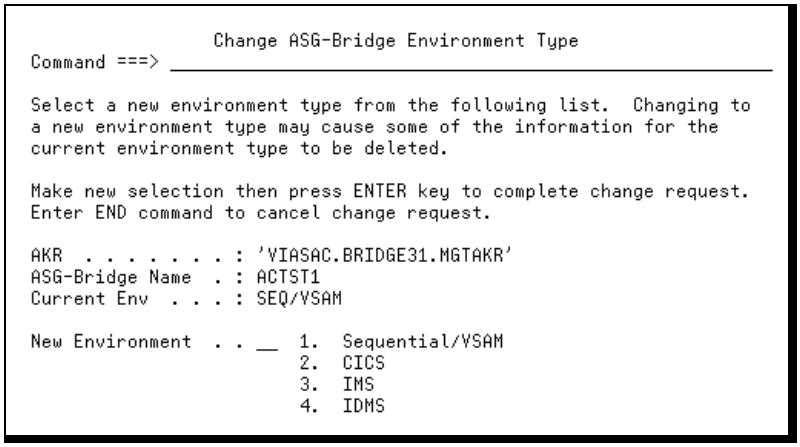
Environment	Bridge/ Pad Field	Record Type	Key/Alt Key	Fixed Array	Segment	DBD Info	Fixed RECL
Sequential VSAM	Yes	Yes	Yes	Yes	No	No	No
CICS	Yes	Yes	Yes	Yes	No	No	Yes
IMS Non-GSAM	Yes	Yes	No	Yes	No	Yes	No
IMS GSAM	Yes	No	No	Yes	Yes	Yes	No
IDMS	Yes	No	No	Yes	No	No	No

To change a Bridge Definition environment

1. Select View ▸ ASG-Bridge Definition Directory from the Bridge Primary screen and press Enter. The ASG-Bridge Definition Directory screen displays.

2. Type `C` to select the Bridge Definition you want to change and press Enter. The Change ASG-Bridge Environment Type pop-up displays (see Figure 96 on page 87).

Figure 96 • Change ASG-Bridge Environment Type Pop-up



3 Select the new environment and press Enter to return to the View - ASG-Bridge Definition Directory screen.

Note:

If you are changing to an environment that requires attribute information, the ASG-Bridge Definition - Attributes screen displays so you can specify the required attributes. Specify the required attribute information and press PF3 to return to the View - ASG-Bridge Definition Directory screen.

Modifying Bridge Definition Attributes

You can change the Bridge Definition source or the file conversion status specified on the ASG-Bridge Definition - Attributes screen. You can also reload the original source that was specified when the Bridge Definition was originally created. You might want to use this feature if you made changes to the original source and you want the Bridge to recognize the changes. You can also specify whether or not you want to resynchronize any previous source record definitions.

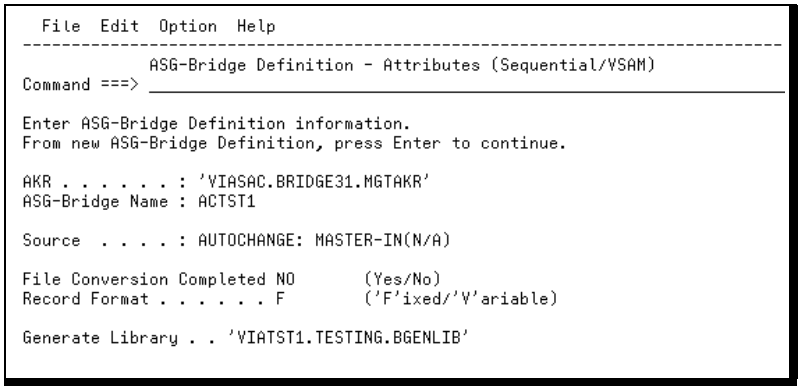
If you change these attributes after generating the Bridge Routine from the original Bridge Definition, you must regenerate the Bridge Routine to incorporate the changes.

To reload original source

1 Select View ▸ ASG-Bridge Definition Directory from the Bridge Primary screen and press Enter. The ASG-Bridge Definition Directory screen displays.

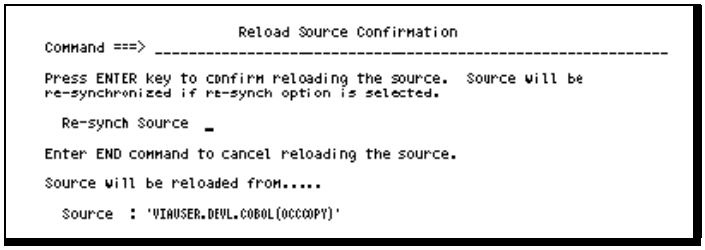
2 Type A to select the Bridge Definition you want to change and press Enter. The ASG-Bridge Definition - Attributes screen displays (see Figure 97).

Figure 97 • ASG-Bridge Definition - Attributes Screen



3 Select File ▸ Reload Source and press Enter. The Reload Source Confirmation pop-up displays (Figure 98).

Figure 98 • Reload Source Confirmation Pop-up



4 Specify if you want to re-synch any previous source record definitions.

5 Press Enter to confirm the reload.

Or

Press PF3 to cancel the reload request and return to the ASG-Bridge Definition - Attributes screen.

Source Reload Errors from Bridge Definition - Attributes Screen

If Bridge is unable to identify the language of the source record you are reloading, finds syntax errors in the source record, or cannot resynchronize source record definitions, these pop-ups display (figures are shown in the section "Source Reload Errors from Bridge Definition - Source Record screen" on page 57). Select an option and press Enter to continue.

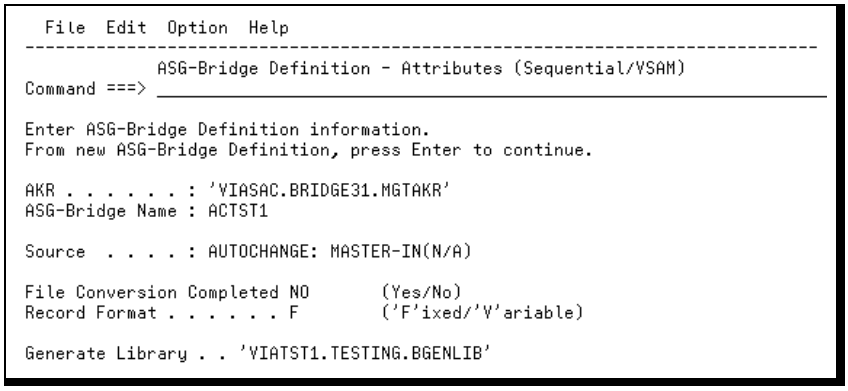
Problem	Pop-up	Options
Cannot identify language	Language Specification (Figure 59 on page 57)	Cancel processing or specify COBOL or PL/I.
Syntax errors	Source Processing Error (Figure 60 on page 57)	Cancel processing or process as different language, or edit the source record.
Cannot resynchronize	Re-synchronizing Source Failed (Figure 61 on page 58)	Cancel processing or accept partial resynchronization.

To change the source

1. Select View ▸ ASG-Bridge Definition Directory from the Bridge Primary screen and press Enter. The ASG-Bridge Definition Directory screen displays.

2. Type `A` to select the Bridge Definition you want to modify and press Enter. The ASG-Bridge Definition - Attributes screen displays (see Figure 99).

Figure 99 • ASG-Bridge Definition - Attributes Screen



3 Select File ▸ Select Source and press Enter. The procedure can vary depending on whether you have the IDMS option installed. This table describes how to proceed with or without installation of the IDMS option:

If Bridge IDMS option is	Then pop-up displayed is…	And perform this action…
Not installed	Select - PDS Member pop-up (Figure 46 on page 48)	Type the name of the dataset. Use the S action to select a member and press Enter. **Note:** Type `B` to browse the member source code before selecting it. Press PF3 to return to the Select - PDS Member pop-up.
Installed - Changing source for a non-IDMS Definition	ASG-Bridge Definition - Copybook Source (Figure 45 on page 47)	Select the PDS option to specify the source dataset and member. The Select - PDS Member pop-up displays. Type the name of the dataset. Type `S` to select a member and press Enter. **Note:** Type `B` to browse the member source code before selecting it. Press PF3 to return to the Select - PDS Member pop-up.
Installed - Changing source for a non-IDMS Definition with IDMS source information	ASG-Bridge Definition - Copybook Source (Figure 45 on page 47)	Select the IDMS Data Dictionary Information option to specify non-IDMS records stored in a Data Dictionary. The ASG-Bridge Definition - IDMS Information pop-up displays.

If Bridge IDMS option is	Then pop-up displayed is…	And perform this action…
		Specify any IDMS parameters and DD information to properly invoke IDMS. Use the I or D action to insert or delete DDs. Type `NEXT` on the command line and press Enter. The ASG-Bridge Definition - Non-IDMS Record Names pop-up (Figure 49 on page 49) displays. Deselect the current record and select a new record. Press PF3.
Installed - Changing source for an IDMS Bridge Definition	ASG-Bridge Definition - IDMS Information pop-up (Figure 50 on page 50)	Specify schema name and version information. Specify any IDMS parameters and DD information to properly invoke IDMS. Use the I or D action to insert or delete DDs. Type `NEXT` on the command line and press Enter. The ASG-Bridge Definition - IDMS Record Names pop-up (Figure 51 on page 51) displays. Select one or more records and press PF3.

4 The Re-synch Option pop-up displays. Type a non-blank character in the Re-synch Source field if you want to resynchronize source record definitions.

5 Press Enter to reload the source and return to the ASG-Bridge Definition - Attributes screen.

Note:

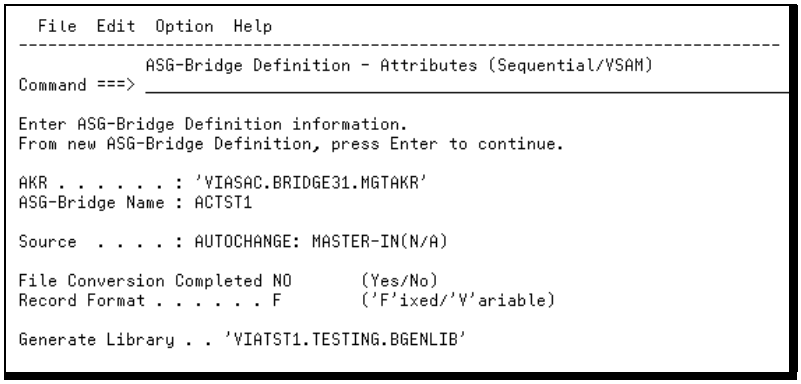
If Bridge encounters any problems identifying the language of the source record you are loading, finds some syntax errors, or cannot resynchronize source record definitions, see "Source Reload Errors from Bridge Definition - Source Record screen" on page 57.

To modify File Conversion Completed status

1 Select View ▸ ASG-Bridge Definition Directory from the Bridge Primary screen and press Enter. The ASG-Bridge Definition Directory screen displays.

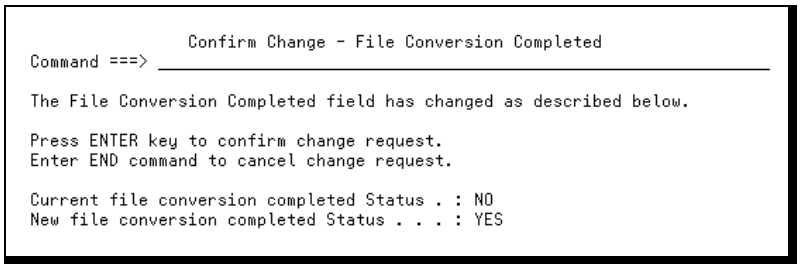
2 Type A to select the Bridge Definition you want to modify and press Enter. The ASG-Bridge Definition - Attributes screen displays (see Figure 100).

Figure 100 • ASG-Bridge Definition - Attributes Screen



3 Modify the status of File Conversion Completed and press PF3. The Confirm Change - File Conversion Completed pop-up displays (see Figure 101).

Figure 101 • Confirm Change - File Conversion Completed Pop-up



4 Verify the information and press Enter to confirm the change or press PF3 to cancel the change and return to the ASG-Bridge Definition - Attributes pop-up.

Note:

The File Conversion Completed specification indicates to Bridge whether the file is converted and determines when forward or reverse bridging is enabled. (see "Reverse Bridging for All Environments" on page 171.)

Deleting a Bridge Definition

You can delete existing Bridge Definitions. Deleting a Bridge Definition does not delete a Bridge Routine that was generated from that Bridge Definition.

To delete an existing Bridge Definition

1. Select View ▸ ASG-Bridge Definition Directory from the Bridge Primary screen and press Enter. The ASG-Bridge Definition Directory screen displays.

2. Type `D` to select a Bridge Definition for deletion and press Enter. The Confirm Delete - ASG-Bridge Definition pop-up displays (see Figure 102).

Figure 102 • Confirm Delete - ASG-Bridge Definition Pop-up

3. Confirm or cancel the delete request and return to the View - ASG-Bridge Definition Directory screen.

AutoChange Bridge Definitions

If you use AutoChange, you can create simple Bridge Definitions that consist of these basic definition elements:

- Source displays as:

  ```
  AUTOCHANGE: <record name from source>
  ```

- Group — Bridge Definitions from AutoChange are automatically assigned to a group named AutoChange.
- Description — The description “Generated from AutoChange” is automatically assigned to Bridge Definitions from AutoChange. The release number and maintenance level of AutoChange is also included in the description.
- Bridge Fields — Bridge Fields are defined in AutoChange with the same Bridge Rules used in Bridge.

The AutoChange Bridge Definitions are listed on the Bridge Definition Directory. You must add any necessary environment and record typing information to complete the definitions and generate Routines (see "Bridge Definition Directory" on page 80).

6 Generating Executable Bridge Routines

This chapter describes how to generate executable Bridge routines and contains these sections:

After you complete the Bridge Definitions you can generate the executable Bridge Routines. You can generate Bridge Definitions from these four areas:

- The ASG-Bridge Primary screen (see Figure 7 on page 14)
- The ASG-Bridge Definition - Source Record screen (see Figure 72 on page 67)
- The View - ASG-Bridge Definition Directory screen (see Figure 105 on page 97)
- The ASG-Bridge Definition - Attributes screen (see Figure 93 on page 84)

Note:

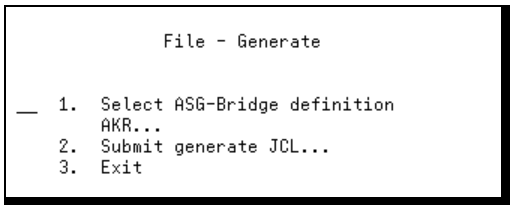
In most cases, when you change a Bridge Definition you must regenerate it to incorporate the changes into the Bridge Routine.

You can generate the current Bridge Definition directly from the ASG-Bridge Definition - Source Record screen and the ASG-Bridge Definition - Attributes screen, or you can select a definition from the list on the View - ASG-Bridge Definition Directory screen.

To generate from the ASG-Bridge Definition - Source Record or the ASG-Bridge Definition - Attributes screen

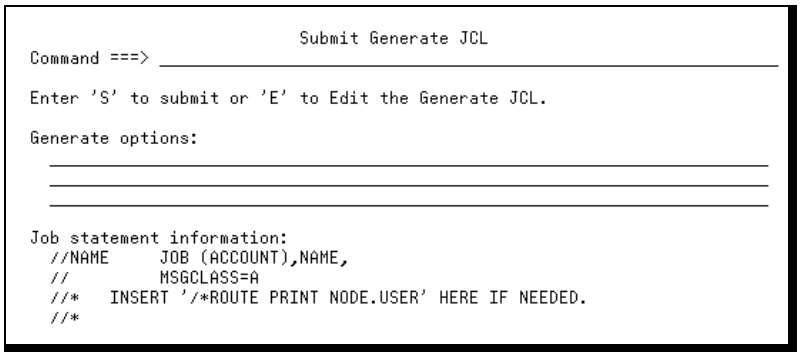
1 Select File ▸ Generate from the Bridge Primary screen and press Enter. The File - Generate pop-up displays (see Figure 103).

Figure 103 • File - Generate Pop-up



2 Select Submit generate JCL to submit the generate request on the Submit Generate JCL pop-up (see Figure 104).

Figure 104 • Submit Generate JCL Pop-up



3 Either specify the job statement information or verify that the existing information is correct.

4 Type `S` on the command line and press Enter to submit the Generate JCL.

Or

Type `E` on the command line and press Enter to edit the Generate JCL. Edit the JCL and type the TSO SUBMIT command on the command line.

5 Press Enter to submit the JCL. Press F3 to return to the Submit Generate JCL pop-up.

6 Press PF3 to return to the screen where you accessed the Submit Generate JCL pop-up.

To generate a definition selected from the View - ASG-Bridge Definition Directory screen

1 Select View ▸ ASG-Bridge Definition Directory and press Enter. The View - ASG-Bridge Definition Directory displays (see Figure 105).

Figure 105 • View - ASG-Bridge Definition Directory screen

```
                    View - ASG-Bridge Definition Directory
Command ===> _____________________________________________ Scroll ===> CSR

AKR 'VIASAC.BRIDGE30.MGTAKR'                                      1 of 215

  A=Attributes    C=Change Environment   K=Check    D=Delete    E=Export
  V=Record View   I=ASG-Bridge Info      O=Copy     G=Generate  R=Rename

  Bridge   Env                                                          >
  Name     Type     Gen Record Layout Source                            >
- -------- -------- --- ------------------------------------------------>
_ A        IDMS     YES IDMS SCHEMA: EMPSCHM, VERSION: 300
_ AAAA     SEQ/VSAM NO  AUTOCHANGE: YYYY-RECORD(N/A)
_ ABC      IDMS     NO
_ ABCD     SEQ/VSAM NO  AUTOCHANGE: MASTER-IN(N/A)
_ ABCDE    SEQ/VSAM NO
_ ABCDEF   SEQ/VSAM NO
_ ABCDEFG  SEQ/VSAM NO
_ ACCTREC  SEQ/VSAM NO
_ ACTEST1  CICS     NO  AUTOCHANGE: WS-OUTLINE-OUTP(N/A)
_ ADVV415A CICS     YES VIASGS.USER.LIB(ADVV415A)
  ATEST    SEQ/VSAM NO
```

2 Type G to select the Bridge Definitions you want to generate and press Enter. The Submit Generate JCL pop-up displays (see Figure 106).

Figure 106 • Submit Generate JCL Pop-up

```
                          Submit Generate JCL
Command ===> ____________________________________________________________

Enter 'S' to submit or 'E' to Edit the Generate JCL.

Generate options:
  ____________________________________________________________________
  ____________________________________________________________________
  ____________________________________________________________________

Job statement information:
  //NAME     JOB (ACCOUNT),NAME,
  //         MSGCLASS=A
  //*   INSERT '/*ROUTE PRINT NODE.USER' HERE IF NEEDED.
  //*
```

3 Type `S` on the command line and press Enter to submit the Generate JCL.

Or

Type `E` on the command line and press Enter to Edit the Generate JCL. Edit the JCL, type `SUBMIT` on the command line, and press Enter to submit the JCL.

4 Press F3 after submission to return to the Submit Generate JCL pop-up.

5 Press PF3 to return to the screen where you accessed the Submit Generate JCL pop-up.

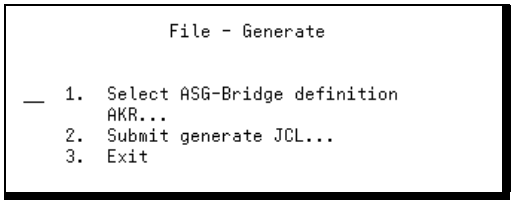
Generating Definitions from a Specific AKR, Group, or Environment

To generate Bridge Definitions from a specific AKR, or to select a group or an environment within an AKR, generate from either the ASG-Bridge Primary screen or the View - ASG-Bridge Definition Directory screen. After you choose the AKR containing the definitions or the group of definitions you want to generate, Bridge presents a selection list that contains only the Bridge Definitions that are ready for generation.

To generate Bridge Definitions from another AKR or narrow the list of Bridge Definitions to a specific group

1 Select File ▸ Generate from the ASG-Bridge Primary screen and press Enter. The File - Generate pop-up displays (see Figure 107).

Figure 107 • File - Generate Pop-up



2 Choose Select ASG-Bridge Definition AKR to designate the AKR you want to generate from and press Enter. The Generate - AKR Selection pop-up displays (see Figure 108 on page 99).

Figure 108 • Generate - AKR Selection Pop-up

```
                        Generate - AKR Selection
Command ===> _____________________________________________ Scroll ===> CSR

Select an AKR then press Enter to continue.                      1 of 1

  AKR data set name
- ---------------------------------------------
_ VIASAC.BRIDGE31.MGTAKR
*************** BOTTOM OF DATA ***************
```

3 Select an AKR and press Enter. The Generate - Selection List pop-up displays (see Figure 109).

Figure 109 • Generate - Selection List Pop-up

```
                Generate - Selection List


 Select to generate
 1   1.  by ASG-Bridge Definitions...
     2.  by Groups...
     3.  by Environment...
     4.  Exit
```

Select Bridge Definitions from a list of definitions in the selected AKR, from a specified group, or from a specified environment.

4 Select the By Bridge Definitions option to choose the definitions you want to generate from a list of Bridge Definitions that are ready for generation. The Generate - ASG-Bridge Definition Selection pop-up displays (see Figure 110).

Figure 110 • Generate - ASG-Bridge Definition Selection Pop-up

```
                Generate - ASG-Bridge Definition Selection
Command ===> _____________________________________________ Scroll ===> CSR

Select one or more ASG-Bridge Definitions.                       1 of 1

AKR : 'VIASAC.BRIDGE31.MGTAKR'

  Name     Environment  Group                          Generate Library >
- -------- ------------ ------------------------------ ----------------->
/ MSTSCM   IDMS         BRIDMSIVP                      VIATST1.TESTING.BG
***************************** BOTTOM OF DATA ******************************
```

5 Select the Bridge Definitions for generation. Press PF3 to save the selections and return to the Generate - Selection List pop-up.

Or

Select the by Groups option from the Generate - Selection List pop-up to choose the group containing the definitions you want to generate and press Enter. The Generate - Group Selection pop-up displays (see Figure 111).

Figure 111 • Generate - Group Selection Pop-up

```
                        Generate - Group Selection
Command ===> _____________________________________________ Scroll ===> CSR

Select a Group then press Enter to choose ASG-Bridge              1 of 1
Definitions.

AKR : 'VIASAC.BRIDGE31.MGTAKR'

  Group
- --------------------------------
_ BRIDMSIVP
********* BOTTOM OF DATA *********
```

6 Select a Group and press Enter. The Generate - ASG-Bridge Definition Selection pop-up displays listing the definitions in the group that are ready for generation.

7 Select the definitions you want to generate. Press PF3 to save the selections and return to the Generate - Group Selection pop-up. Press PF3 to return to the Generate - Selection List pop-up.

Or

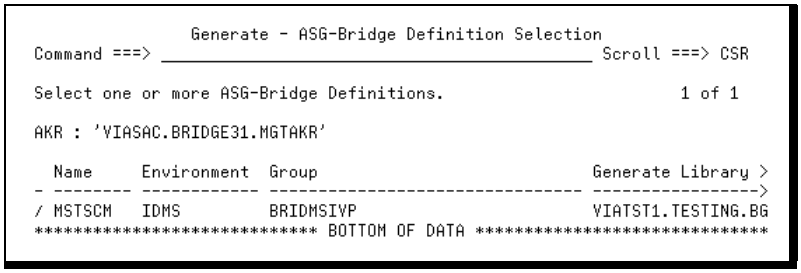
Select the by Environment option to choose the environment of the definitions you want to generate. The Generate - Environment Selection pop-up displays (see Figure 112).

Figure 112 • Generate - Environment Selection Pop-up

```
                    Generate - Environment Selection
Command ===> _____________________________________________ Scroll ===> CSR

Select an Environment then press Enter to choose ASG-Bridge       1 of 1
Definitions.

AKR : 'VIASAC.BRIDGE31.MGTAKR'

  Environment
- --------------------------------
_ IDMS
********* BOTTOM OF DATA *********
```

8 Select an Environment and press Enter. The Generate - ASG-Bridge Definition Selection pop-up displays, listing the definitions in the environment that are ready for generation (see Figure 113).

Figure 113 • Generate - ASG-Bridge Definition Selection Pop-up



9 Select the definitions you want to generate. Press PF3 to save the selections and return to the Generate - Environment Selection pop-up.

10 Press PF3 again to return to the Generate - Selection List pop-up.

11 Press PF3 to return to the Generate AKR Selection pop-up. From here you can choose a different AKR to generate more definitions by repeating the process from step 3 on page 99, or proceed to step 12.

12 Press PF3 to return to the File - Generate pop-up. Select Submit Generate JCL to submit the generate request on the Submit Generate JCL pop-up.

13 Type `S` on the command line and press Enter to submit the Generate JCL.

Or

Type `E` on the command line and press Enter to edit the Generate JCL. Edit the JCL, type `SUBMIT` on the command line, and press Enter to submit the JCL.

14 Press F3 after submission to return to the Submit Generate JCL pop-up.

15 Press PF3 to return to the screen where you accessed the File - Generate pop-up.

Generating IDMS Subschema Conversion Routines (SCRs)

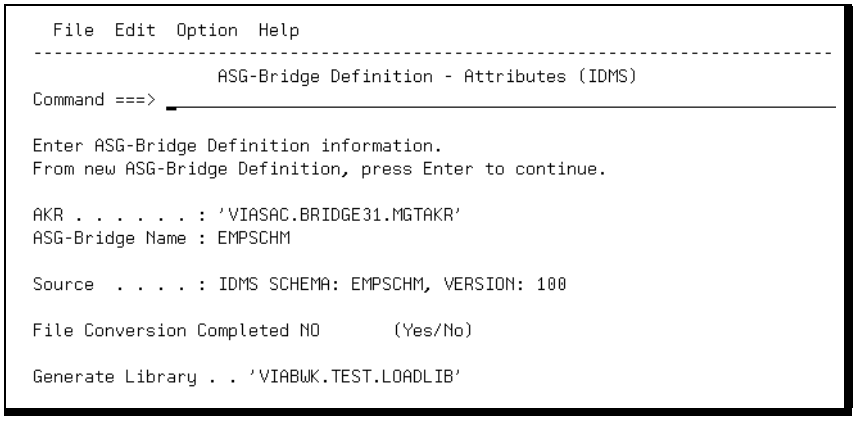
When you change or regenerate an IDMS Bridge Definition, you must generate a Subschema Conversion Routine (SCR) for each subschema after the Bridge Routine is generated. (See "Integrating Bridge into IDMS Programs" on page 163 for more information.)

To generate an IDMS subschema conversion routine

1 Select View ▸ ASG-Bridge Definition Directory from the Bridge Primary screen and press Enter. The View - ASG-Bridge Definition Directory screen displays.

2 Type A to select the IDMS Bridge Definition and press Enter. The ASG-Bridge Definition - Attributes screen displays (see Figure 114).

Figure 114 • ASG-Bridge Definition Attributes screen



3 Type the name of the generate library in the Generate Library field.

Or

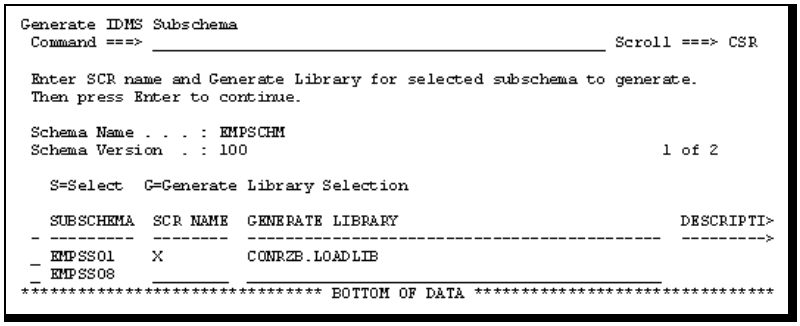
Select File ▸ Select Generate Library.

Press Enter. The Associate - Generate Library pop-up displays with a list of generate libraries.

4 Select a library and press PF3 to save and return to the ASG-Bridge Definition - Attributes screen.

5 Select File ▸ Generate and press Enter. The Generate IDMS Subschema screen displays (see Figure 115 on page 103).

Figure 115 • Generate IDMS Subschema screen



6 Specify the generate library by typing the SCR library name in the Generate Library field.

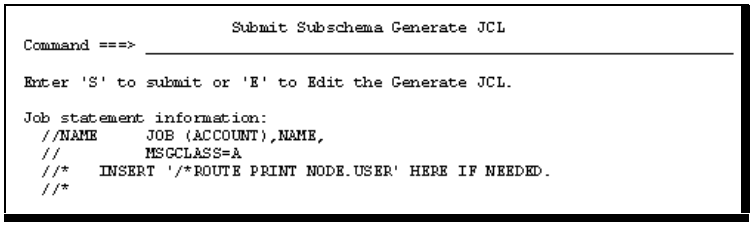
Or

Type `G` to select the Generate Library and press Enter. The Associate Generate Library pop-up displays.

7 Select a generate library and press Enter to return to the Generate IDMS Subschema.

8 Select the subschemas to generate and press Enter. The Submit Subschema Generate JCL pop-up displays (see Figure 116).

Figure 116 • Submit Subschema Generate JCL screen



9 Type `S` on the command line and press Enter to submit the Generate JCL.

Or

Type `E` on the command line and press Enter to Edit the Generate JCL. Edit the JCL, type `SUBMIT` on the command line, and press Enter to submit the JCL.

10 Press F3 to return to the Submit Generate JCL pop-up.

11 Press PF3 to return to ASG-Bridge Definition - Attributes screen.

Maintaining Generate Libraries

The Generate Library is a load module library that stores generated Bridge Routines. From here you can view a list of current Generate Libraries or library members, delete a library association with a Bridge Definition, modify the description of a Generate Library, specify new Generate Libraries to Bridge, or delete a Generate Library.

Note:

The Generate Library must be an existing load module library.

To view Bridge Generate Libraries

1 Select View ▸ Bridge Generate Libraries from the Bridge Primary screen and press Enter. The View - ASG-Bridge Generate Libraries screen displays (see Figure 117).

Figure 117 • View - ASG-Bridge Generate Libraries screen

```
  File  Option  Help
--------------------------------------------------------------------------------
                    View - ASG-Bridge Generate Libraries
Command ===> _____________________________________________ Scroll ===> CSR

ASG-Bridge Mgmt AKR: 'VIASAC.BRIDGE30.MGTAKR'                        1 of 74

  M=Modify  D=Delete  S=Select

  Generate Library data set name               Mbrs
- ---------------------------------------------- ----
_ CONESP1.EURO.LOADLIB                           10
_ CONRXB.BRIDGE.GENLIB                            6
_ CONRXB.LOADLIB2                                 2
_ CONRXB.TEST.LOADLIB                             2
_ CONRZB.LOADLIB                                  3
_ CONVXA.LOADLIB1                                 1
_ VIAARJ.EB.LOADLIB                               4
_ VIAAXE.BRIDGE.TEST.GENLIB                       4
_ VIAAXL.TRN.LOADLIB                              2
_ VIAAXR.TEST.LOAD                                4
_ VIABR31.DEVL.XALOAD                             9
_ VIABR32.DEVL.XALOAD                             0
_ VIABWK.TEST.LOADLIB                            35
```

2 Press PF3 to return to the Bridge Primary screen.

To specify a new Bridge Generate Library to Bridge

1 Select File ▸ New from the Bridge Primary screen and press Enter. The File - New pop-up displays (see Figure 118).

Figure 118 • File - New Pop-up

```
                File - New


Select to make new
1   1.  ASG-Bridge Parameters...
    2.  ASG-Bridge Rules...
    3.  ASG-Bridge Definition...
    4.  ASG-Bridge Generate Libraries...
    5.  Exit
```

Or

Select View ▸ ASG-Bridge Generate Libraries and press Enter. The View - ASG-Bridge Generate Libraries screen displays.

2 Select File ▸ New ASG-Bridge Generate Library and press Enter. The File - New ASG-Bridge Generate Library pop-up displays (see Figure 119).

Figure 119 • File - New ASG-Bridge Generate Library Pop-up

```
                File - New ASG-Bridge Generate Library
Command ===> ____________________________________________________________

Enter the Generate Library information, then press Enter.

Generate Library Name   ____________________________________________
Description . . . . . . ____________________________________________
                        ____________________________________________
                        ____________________________________________
```

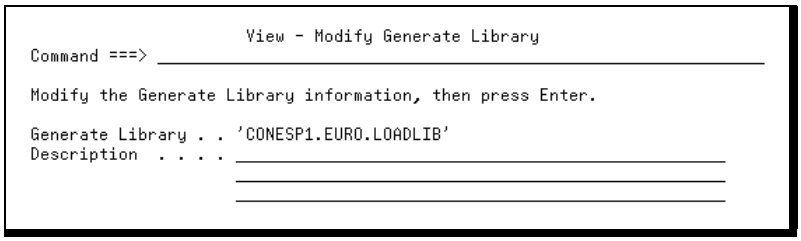
3 Type the Generate Library name and description in the appropriate fields and press PF3 to save and return to the Bridge Primary screen or the View - ASG-Bridge Generate Libraries screen.

To modify the description of a Generate Library

1 Select View ▸ Bridge Generate Libraries from the Bridge Primary screen and press Enter. The View - ASG-Bridge Generate Libraries screen displays.

2 Type `M` to select the library you want to modify and press Enter. The View - Modify Generate Library pop-up displays (see Figure 120).

Figure 120 • View - Modify Generate Library Pop-up



3 Make the modifications to the Generate Library Description field and press PF3 to save the changes and to return to the View - ASG-Bridge Generate Libraries screen.

To delete a Generate Library

1 Select View ▸ Bridge Generate Libraries from the Bridge Primary screen and press Enter. The View - ASG-Bridge Generate Libraries screen displays.

2 Type `D` to select a library for deletion and press Enter. The Confirm Delete - Generate Library pop-up displays (see Figure 121).

Figure 121 • Confirm Delete - Generate Library Pop-up

3 Press Enter to confirm or PF3 to cancel the delete request and to return to the View - ASG-Bridge Generate Libraries screen.

Note:

Deleting a Generate Library removes the library from the list of available Generate Libraries. Bridge Routines in the library are still available to Bridge.

To view Bridge Generate Library members and member attributes

1. Select View ▸ Bridge Generate Libraries from the Bridge Primary screen and press Enter. The View - ASG-Bridge Generate Libraries screen displays.

2. Type `S` to select a Generate Library and press Enter. The View - ASG-Bridge Generate Library Members screen displays listing the members of the selected library (see Figure 122).

Figure 122 • View - Bridge Generate Library Members screen

```
  File  Option  Help
--------------------------------------------------------------------------------
                View - ASG-Bridge Generate Library Members
Command ===> _____________________________________________ Scroll ===> CSR

Generate Library  'CONESP1.EURO.LOADLIB'                          1 of 10

  D=Delete Association   A=Attributes

  Bridge    Assoc.       Assoc.                                            >
  Name      Date         Time      AKR Name                                >
- --------  -----------  --------  ----------------------------------------->
_ EMPSCHM   11-MAY-2000  14:15:50  VIASAC.BRIDGE30.MGTAKR
_ EUROESP   28-APR-1999  02:26:05  CONESP1.GERARDO.AKR
_ EUROTESP  28-APR-1999  01:54:19  CONESP1.GERARDO.AKR
_ EUROTPTS  28-APR-1999  02:37:45  CONESP1.GERARDO.AKR
_ EURO2PTS  28-APR-1999  04:34:13  CONESP1.GERARDO.AKR
_ EURO3PTS  28-APR-1999  07:43:46  CONESP1.GERARDO.AKR
_ IMPORTS   13-JUN-2001  09:18:26  VIASAC.BRIDGE31.MGTAKR
_ PAYCHECK  13-JUN-2001  08:34:04  VIASAC.BRIDGE31.MGTAKR
_ PAYCHKS3  12-JUN-2001  09:42:20  VIASAC.BRIDGE31.MGTAKR
_ PAYCHKS4  12-JUN-2001  09:53:05  VIASAC.BRIDGE31.MGTAKR
******************************** BOTTOM OF DATA ********************************
```

3. Type `A` to select a member if you want to view or modify the Bridge Definition attributes on the ASG-Bridge Definition - Attributes screen and press Enter.

4. Press PF3 to return to the View - ASG-Bridge Generate Library Members screen.

5. Press PF3 to return to the View - ASG-Bridge Generate Libraries screen.

To delete the Bridge Definition association with a Generate Library

1. Select View ▸ Bridge Generate Libraries from the Bridge Primary screen and press Enter. The View - ASG-Bridge Generate Libraries screen displays.

2. Type `D` to select a definition to delete the association and press Enter. The Confirm Delete - Generate Library pop-up displays.

3 Press Enter to confirm the delete request.

Or

`Type END` to cancel the request.

4 Press PF3 to return to the View - ASG-Bridge Generate Libraries screen.

7 Cross-Reference Feature

This chapter describes how you can view a summary of all of Bridge Definitions within a specified AKR along with their respective attributes and contains these sections:

The View - ASG-Bridge Definition Cross Reference screen displays a summary view of all of the Bridge Definitions in a specified AKR along with their respective attributes. You can choose from a list of sort preferences to display the information in various sort orders and you can also print the view.

Use the primary FIND command to search for and to select items on this screen. Use the FIRST, LAST, NEXT, PREVIOUS, ALL, or APPEND operands with the FIND command to refine the search.

Use the primary command RESET to unselect items you selected by using the FIND command.

Note:

You can use the primary FIND command on all list screens and pop-ups. See "Common Commands," on page 243 for a list of commands common to all Bridge screens.

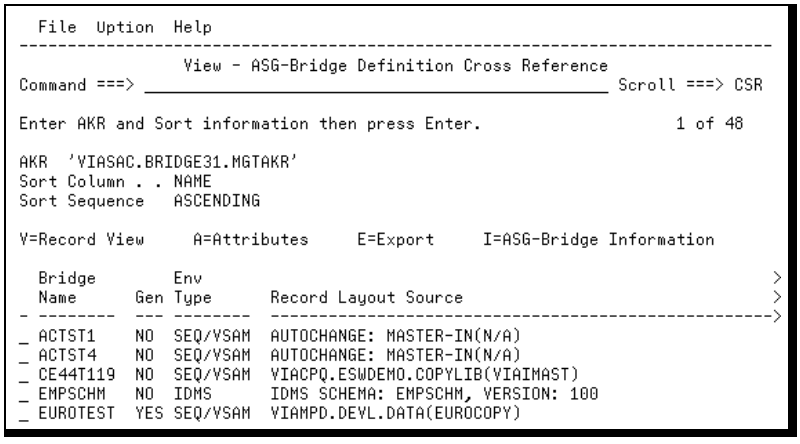
These are the tasks you can access from the View - Bridge Definition Cross Reference screen:

Task	Action
Define Source Record information	Select a Bridge Definition with the V action. See "Defining Source Record Information" on page 54.
Review or modify definition attributes	Select a Bridge Definition with the A action. See "To name a new Bridge Definition and specify definition attributes" on page 43.
Export Bridge Definitions.	Select a Bridge Definition with the E action. See "Importing/Exporting Parameters, Rules, and Definitions" on page 115.
View statistical Bridge Definition information	Select a Bridge Definition with the I action. See "Viewing Bridge Definition Summary Information" on page 79.

To view a cross-reference of Bridge Definitions

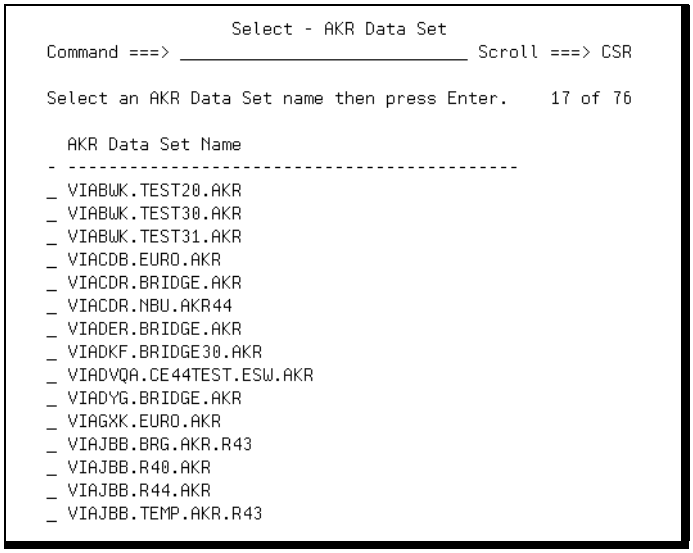
1 Select View ▸ Cross References from the Bridge Primary screen and press Enter. The View - ASG-Bridge Definition Cross Reference screen displays showing the Bridge Definitions in the currently specified AKR (see Figure 123).

Figure 123 • View - ASG-Bridge Definition Cross Reference screen



To view the Bridge Definitions in a different AKR, select File ▸ Select AKR and press Enter. Select an AKR from the Select - AKR Data Set pop-up and press Enter (see Figure 124 on page 111). Alternately, you can type the name of an AKR in the AKR field on the View - Bridge Definition Cross Reference screen and press Enter.

Figure 124 • Select - AKR Data Set Pop-up



2 Select File ▸ Exit to return to the Bridge Primary screen.

Or

Press PF3 to return to the Bridge Primary screen.

Sorting Information on the View - Bridge Definition Cross Reference Screen

You can display the information on the View - ASG-Bridge Definition Cross Reference screen in various sort orders by selecting from a list of sort preferences or by typing sort preferences directly into Sort Column and Sort Sequence fields.

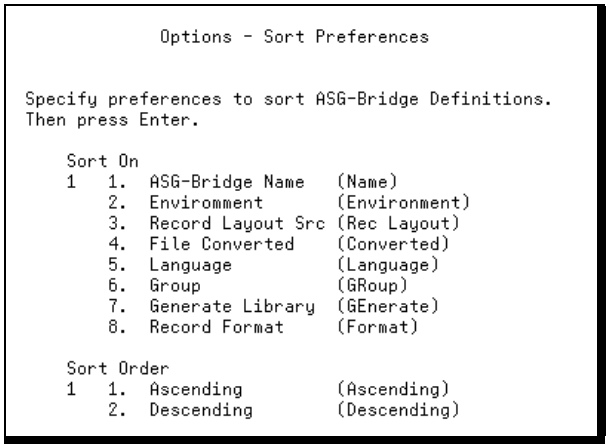
The sort preference selection for the Sort Column determines the primary sort order of the information presented. The Bridge Name is the secondary sort order.

To specify sort preferences on the View - ASG-Bridge Definition Cross Reference screen

1 Select View ▸ Cross References from the Bridge Primary screen and press Enter. The View - ASG-Bridge Definition Cross Reference screen displays.

2 Select Option ▸ Choose Sort Preferences and press Enter. The Options - Sort Preferences pop-up displays (see Figure 125).

Figure 125 • Options - Sort Preferences Pop-up



3 Select the primary key you want to sort on and indicate the sort order.

4 Press Enter to activate the sort and return to the View - ASG-Bridge Definition Cross Reference screen. The information displays according to the preferences you selected.

5 Select File ▸ Exit.

Or

Press PF3 to return to the Bridge Primary screen.

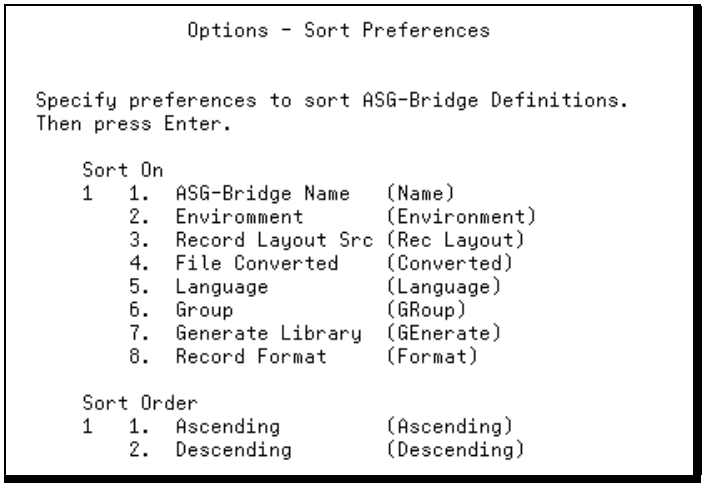
To specify sort preferences from the View - ASG-Bridge Definition Cross Reference screen

1 Select View ▸ ASG-Bridge Definition Cross Reference from the Bridge Primary screen and press Enter. The View - ASG-Bridge Definition Cross Reference screen displays.

2 Type your preferences into the sort information fields on the screen. Use the first few letters of the option as an abbreviation, for example `GE` for Generate or `N` for Name.

3 Type the sort preference or abbreviation. The sort preferences are listed on the Options - Sort Preferences pop-up (see Figure 126).

Figure 126 • Options - Sort Preferences Pop-up



4 Type the sort order (`A` for ascending or `D` for descending) in the Sort Sequence field.

5 Press Enter to initiate the sort and display the information in the new sort order on the View - ASG-Bridge Definition Cross Reference screen.

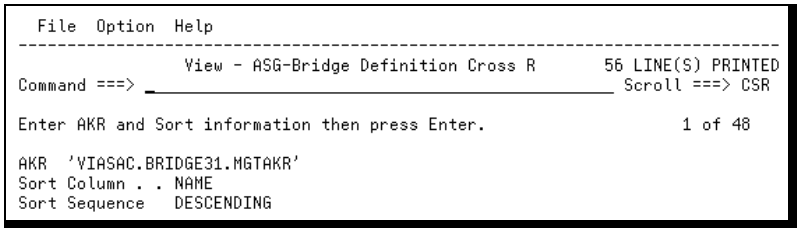
Printing Bridge Definition Cross-Reference Information

You can print the information shown on the View - ASG-Bridge Definition Cross Reference screen.

To print the view

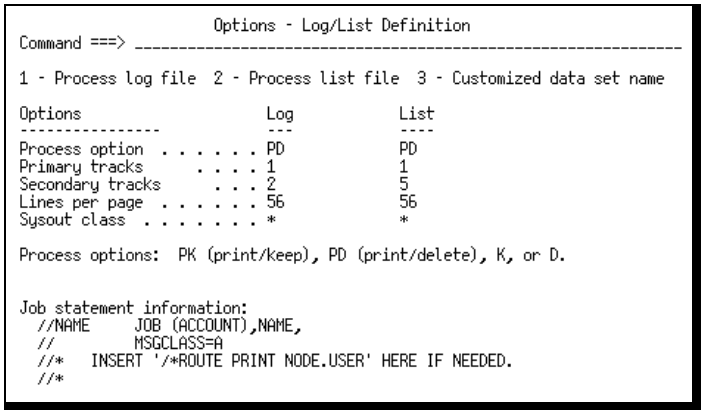
1 Select File ▸ Print and press Enter. A message displays specifying how many lines are printed to the List file (see Figure 127).

Figure 127 • Number of Lines Printed to List File Message



2 Select Option ▸ Process list file to process the list file and press Enter. The Options - Log/List Definition pop-up displays (see Figure 128).

Figure 128 • Options - Log/List Definition Pop-up



3 Specify options and press PF3 to process the list file and to return to the View - ASG-Bridge Definition Cross Reference screen.

4 Press PF3 to return to the Bridge Primary screen.

8 Importing/Exporting Parameters, Rules, and Definitions

This chapter describes how to import and export Bridge Parameters, Rules, and Definitions, and contains these sections:

When enabled, the Bridge Event Log records imports and exports of Bridge Parameters, Rules, and Definitions.

Import/Export Text

A card-image text file imports Bridge Definitions as sequential files or PDS members coded in an English language keyword-based syntax format. Import/Export file statements are either keyword statements, value statements, or comment statements. Import/Export text processors are not usually case-sensitive; however, keywords must be uppercase and the type value is case-sensitive in Bridge Definitions.

Keyword Statements

Keyword statements contain one or more words enclosed by keyword delineation symbols: less-than (<) and greater-than (>) symbols. The keywords identify an attribute or function of a Bridge Definition or serve as delineators within or between Bridge Definitions. Most keyword statements are followed by one or more value statements, for example:

```
<BRIDGE NAME>
```

If a statement displays a keyword statement (it contains keyword delineation characters), and the keyword itself is not identifiable, the import text processor rejects the statement. The Bridge Definition being imported is rejected as well.

Value Statements

Any import text statement that does not contain keyword delineation characters is considered a value statement and is applied to the previous keyword statement then either accepted or rejected accordingly.

Comment Statements

The import text processor ignores comment statements beginning with an asterisk (*).

Import/Export Text Example

This example of import/export text illustrates comment statements, keyword statements, and value statements.

```
*This is a comment statement.
*The following statements are keyword statements followed by
their *respective value statements.
<BRIDGE NAME>
APMASTER
<GROUP>
Accounting
<DESCRIPTION>
This is the Bridge Definition for the Accounts Payable Master
File used in the Headquarters Accounting System
```

Exporting Bridge Parameters, Rules, or Definitions

Bridge exports Bridge Parameters, Rules, or Definitions to the external file that you specify. Reasons to export Bridge Parameters, Rules, or Definitions include:

- To save them in a file for recovery purposes
- To send them to a version control manager
- To copy them to another AKR (production to development or development to production)

When you import or export Bridge Parameters and Rules, you always import them to or export them from the base management AKR.

When you export Bridge Definitions, the export file normally contains the name of the AKR from where the definitions were exported. When you import Bridge Definitions, they are stored in the AKR designated in the export file, unless you specify a different AKR to override the AKR in the export file during the import process.

To export Bridge Parameters or Rules from the View - ASG-Bridge Parameters or View - ASG-Bridge Rules screen

1 Select View ▸ ASG-Bridge Parameters from the Bridge Primary screen and press Enter. The View - ASG-Bridge Parameters screen displays.

Or

Select View ▸ ASG-Bridge Rules from the Bridge Primary screen and press Enter. The View - ASG-Bridge Rules screen displays.

2 Type E to select the parameters or rules for export and press Enter. The File - Export Bridge Parameters (see Figure 129) or File - Export Bridge Rules (see Figure 130) pop-up displays.

Figure 129 • File - Export ASG-Bridge Parameters Pop-up

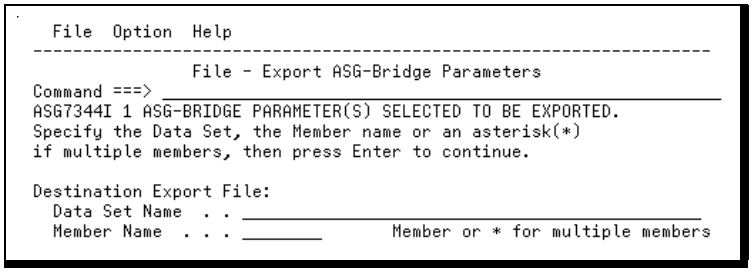
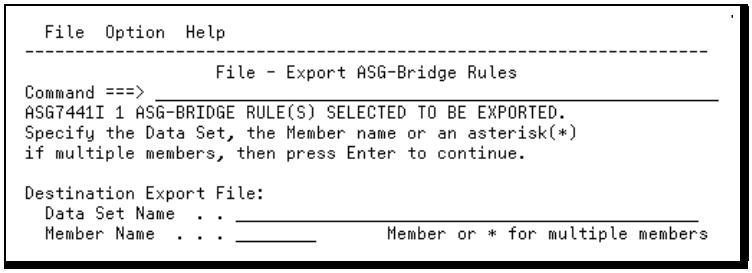


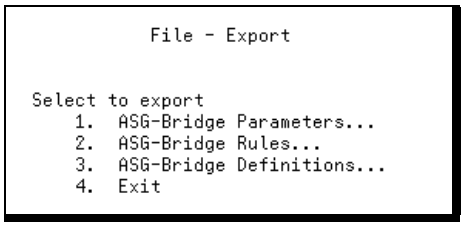
Figure 130 • File - Export ASG-Bridge Rules



To export Bridge Parameters or Rules from the File - Export ASG-Bridge Parameters or File - Export ASG-Bridge Rules pop-up

1 Select File ▸ Export from the Bridge Primary screen and press Enter. The File - Export pop-up displays (see Figure 131).

Figure 131 • File - Export Pop-up



2 Select ASG-Bridge Parameters or ASG-Bridge Rules and press Enter. The File - Export ASG-Bridge Parameters pop-up (Figure 132) or the File - Export ASG-Bridge Rules pop-up displays (Figure 133).

Figure 132 • File - Export ASG-Bridge Parameters Pop-up

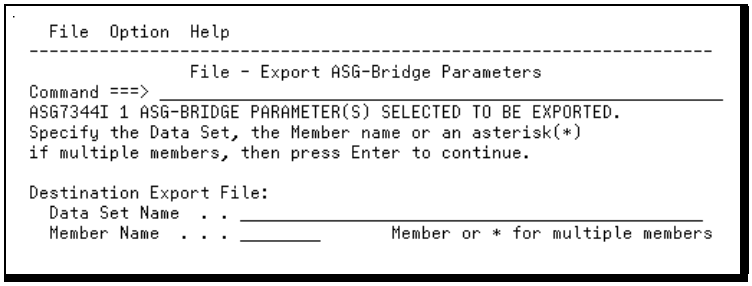
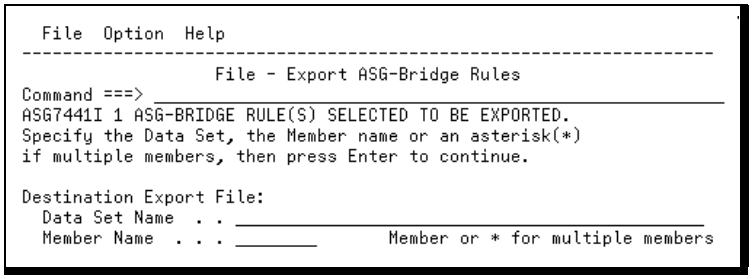


Figure 133 • File - ASG-Export Bridge Rules Pop-up



3 If you have already selected Bridge Parameters or Rules for export, proceed to step 6 on page 120.

4 If you need to select Bridge Parameters, select File ▸ Select Export Bridge Parameters The Select - Bridge Parameters pop-up displays (see Figure 134).

Or

If you need to select Bridge Rules, select File ▸ Export Bridge Rules and press Enter. The Select - ASG-Bridge Rules pop-up displays (see Figure 135).

Figure 134 • Select - ASG-Bridge Parameters Pop-up

```
                Select - ASG-Bridge Parameters
Command ===> _____________________________ Scroll ===> CSR

Select ASG-Bridge Parameters then press          1 of 178
Enter.

  Name      Description
- --------  --------------------------------
_ ANYNEWS   ANY NEWS SCALING FACTOR
_ ANYOLDS   ANY OLDS SCALING FACTOR
_ ATESTB
_ ATEST1    TEST
_ ATSEURO   AUSTRIAN SCHILLING TO EURO CONVE
_ AXEERATE  TEST EXCHANGE RATE
_ BEFEURO   BELGIAN FRANC TO EURO CONVERSION
_ BRDGTSTA  TEST RULE
_ CHECK     CHECK NUMERIC VALUES
_ CONV      NEW CURRENCY
_ CONVERT   OLD  CURRENCY
_ CONVTYPE  CONVERSION TYPE,C->E,E->C,C->C
_ CURFLG    CURRENCY TYPE FIELD
_ CURRENCY  CURRENCY FLAG
```

Figure 135 • Select - ASG-Bridge Rules Pop-up

```
                  Select - ASG-Bridge Rules
Command ===> _____________________________ Scroll ===> CSR

Select ASG-Bridge Rules then press Enter.        1 of 174

  Name      Description
- --------  ------------------------------------------------
_ AA        PRACTICE
_ ACTACZ    NATIONAL CURRENCY TO NATIONAL CURRENCY - GEN
_ ACTBOTH   ANY CURRENCY TO EURO PASS BACK INPUT VALUE
_ ACTEUROZ  ANY CURRENCY TO EURO ZONED WITH ROUNDING LOS
_ ACTEURZN  ANY CURRENCY TO EURO ZONED NO ROUNDING LOSS
_ ADJTEST   ADJUST THE SIZE OF A (FILLER) FIELD
_ ADJUST    ADJUST THE SIZE OF A (FILLER) FIELD
_ ADJUSTL   ADJUST SIZE OF FIELD TO THE LEFT
_ ADJUSTL1  SHIFT DATA 1 BYTE LEFT
_ ADJUSTN   ADJUST THE SIZE OF A (FILLER) FIELD
_ ADJUSTNL  ADJUST THE LENGTH OF NUMERIC FIELDS
_ ANY2EURO  ANY CURRENCY TO EURO
_ AXRTEST   TESTRULE
_ BRFF      BRIDGE FF TO EURO ZONED DECIMAL
```

5 Select parameters or rules for export and press Enter to return to the File - Export ASG-Bridge Parameters or the File - Export ASG-Bridge Rules pop-up.

6 Specify the dataset name for the external export file. You may need to specify or select a member name depending on the export dataset type and how you want to save the definitions.This table describes export dataset types, requirements, and results:

Export Data Set Type	Member Name	Export Results
Sequential	Not Allowed	All Bridge Parameters or Rules are batched together and written to the file.
Partitioned	Allowed	All Bridge Parameters or Rules are batched together and written to the member specified.
Partitioned	asterisk (*)	Each Bridge Parameter or Rule is exported to a separate member that is named the same as the Bridge Definition it contains.

7 Select File ▸ Select Destination Member to specify a member name and press Enter. The Select - PDS Member pop-up displays listing the available members.

8 Select a member and press Enter to return to the File - Export ASG-Bridge Parameters or File - Export ASG-Bridge Rules pop-up.

Or

Type a member name or an asterisk (*) in the Member name field.

9 Press Enter to complete the export. A message displays on the File - Export ASG-Bridge Parameters or File - Export ASG-Bridge Rules pop-up confirming the number of Bridge Parameters or Rules exported.

10 Press PF3 to return to File - Export pop-up.

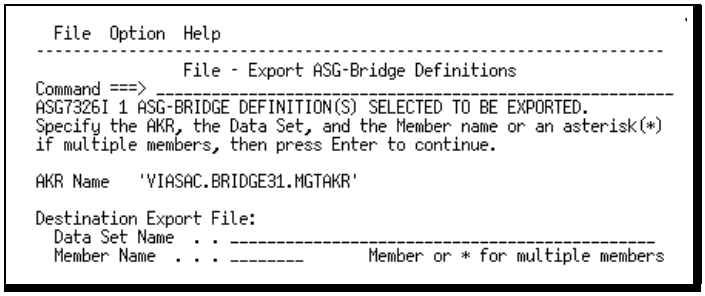
To export Bridge Definitions

1 Use one of these methods to access the File - Export ASG-Bridge Definitions pop-up:

- Select File ▸ Export from the Bridge Primary screen and press Enter. The File - Export pop-up displays. Choose ASG-Bridge Definitions and press Enter. The File - Export ASG-Bridge Definitions pop-up displays.
- Select View - Cross Reference from the Bridge Primary screen and press Enter. The ASG-Bridge Definitions pop-up displays. Select File ▸ Export and press Enter. The File - Export ASG-Bridge Definitions pop-up displays.

- Select View ▸ Bridge Definition Directory or View ▸ Cross References from the Bridge Primary screen and press Enter. The View - ASG-Bridge Definition Directory screen or the View - ASG-Bridge Definition Cross Reference screen displays. Type E to select the Bridge Definitions to export and press Enter. The File - Export ASG-Bridge Definitions pop-up displays (see Figure 136).

Figure 136 • File - Export ASG-Bridge Definition Pop-up

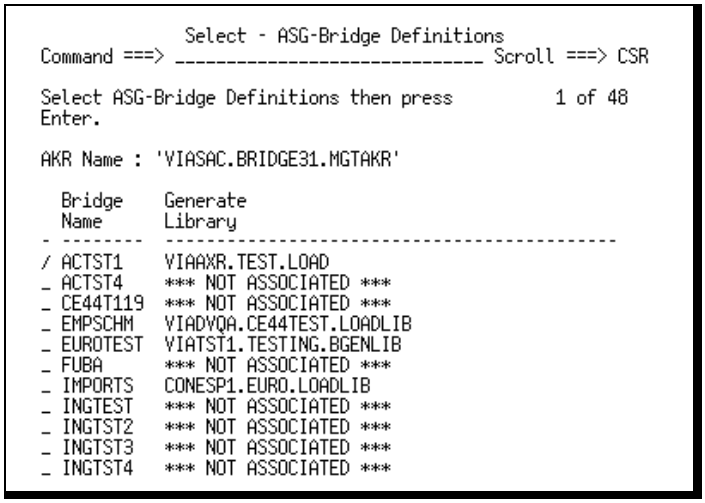


2 If you have already selected Bridge Definitions for export, proceed to step 4 on page 122.

3 If you need to select Bridge Definitions, select File on the action bar and select one of these options:

Option	Description
Select AKR	Select an AKR on the Select - AKR Data Set pop-up that contains Bridge Definitions for export. Press Enter to return to the File - Export ASG-Bridge Definitions pop-up.
Select Export Bridge Definitions	Select Bridge Definitions in the currently specified AKR to export on the Select - Bridge Definitions pop-up (see Figure 137 on page 122). Press Enter to return to File - Export Bridge Definitions pop-up. These commands can be used to select Bridge Definitions: SELECT ALL — Selects all definitions SELECT NONE — Unselects any selected definitions S or / — In the line command area selects a definition Space over an S or / in the line command area to clear a selection

Figure 137 • Select - ASG-Bridge Definitions Pop-up



4 Specify the dataset name for the external export file. You may need to specify or select a member name depending on the export dataset type and specify how you want to save the definitions. These are the export dataset types, requirements, and results:

Export Data Set Type	Member Name	Export Results
Sequential	Not Allowed	All Bridge Definitions are batched together and written to the file
Partitioned	Allowed	All Bridge Definitions are batched together and written to the member specified
Partitioned	asterisk (*)	Each Bridge Definition is exported to a separate member that is named the same as the Bridge Definition it contains

5 Select File ▸ Select Destination Member to specify a member name and press enter. The Select - PDS Member pop-up displays listing the available members. Select a member and press Enter to return to the File - Export ASG-Bridge Definitions pop-up.

Or

Type a member name or an asterisk (*) in the Member name field.

6 Press Enter to complete the export. A message displays on the File - Export ASG-Bridge Definitions pop-up confirming the number of Bridge Definitions exported.

7 Press PF3 to return to the screen where you accessed the File - Export ASG-Bridge Definition pop-up.

Importing Bridge Parameters, Rules, or Definitions

Bridge Parameters, Rules, and Definitions are imported from a user-specified external file. Some reasons to import Bridge Parameters, Rules, or Definitions are:

- Reconstruct a lost or damaged AKR.
- Restore a previous version of a Bridge Parameter, Rule, or Definition.
- Copy Bridge Parameters, Rules, or Definitions from another AKR (production to development or development to production).
- Migrate Bridge Parameters, Rules, or Definitions from an AKR used with a previous Bridge version.
- Import starter sets of Bridge Parameters or Rules.

The Import Facility has a feature that prevents you from accidently overwriting existing Bridge parameters, rules, or definitions when importing definitions with the same name. During the import process you can, however, replace parameters, rules, or definitions with the same name.

The session list file records all import results for historical reference and error analysis. If Bridge encounters syntax or processing errors, the complete Bridge Parameter, Rule, or Definition import text writes to the session list file along with descriptive error messages. You can browse or print the session list dataset to help determine the cause of an import failure or error.

To import Bridge Parameters or Rules

1 Select File ▸ Import from the Bridge Primary screen and press Enter. The File - Import pop-up displays.

2 Select Bridge Parameters or Bridge Rules and press Enter. The File - Import ASG-Bridge Parameters (see Figure 138 on page 124) or File - Import - ASG-Bridge Rules (see Figure 139 on page 124) pop-up displays.

Figure 138 • File - Import ASG-Bridge Parameters Pop-up

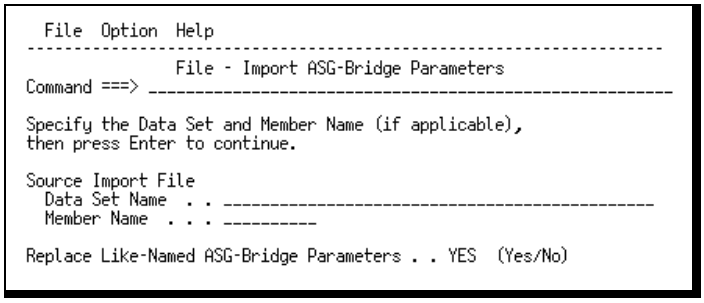
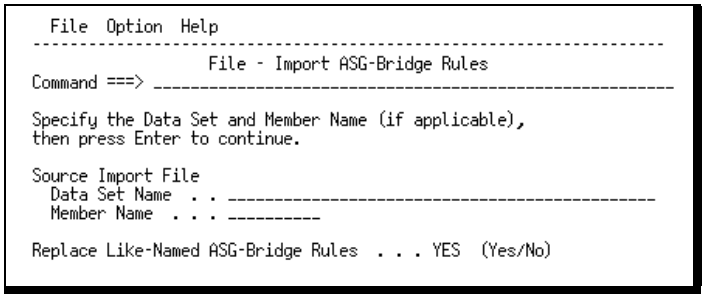


Figure 139 • File - Import ASG-Bridge Rules Pop-up



3 Specify the import file dataset name and member name according to the dataset type. These are the dataset types:

Import Dataset Type	Member Name
Sequential	Not Allowed
Partitioned	Allowed

4 Select File ▸ Select Member to specify a member name and press Enter. The Select - PDS Member pop-up displays listing the available members. Select a member and press Enter to return to the File - Import ASG-Bridge Parameters or File - Import ASG-Bridge Rules pop-up.

Or

Type a member name in the Member Name field.

5 Specify whether to replace like named parameters or rules and press Enter. A message displays verifying the number of parameters or rules imported.

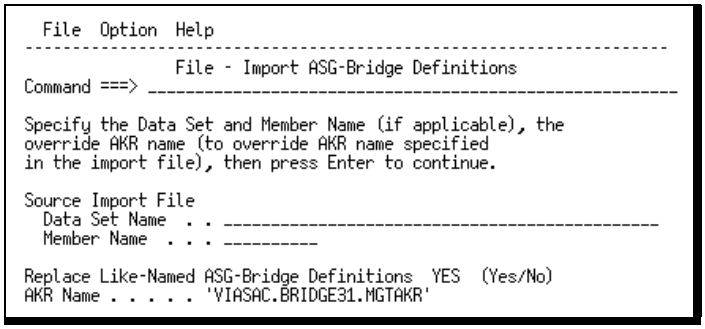
6 Press PF3 to return to the File - Import pop-up.

To import Bridge Definitions

1 Select File ▸ Import from the Bridge Primary screen and press Enter. The File - Import pop-up displays.

2 Select ASG-Bridge Definitions and press Enter. The File - Import ASG-Bridge Definitions pop-up displays (see Figure 140).

Figure 140 • File - Import ASG-Bridge Definitions Pop-up



3 Specify the dataset and/or member name for the source import file according to the dataset type. These are the dataset types:

Import Dataset type	Member Name
Sequential	Not Allowed
Partitioned	Allowed

4 Select File ▸ Select Member to specify a member name and press Enter. The Select - PDS Member pop-up displays listing the available members.

5 Select a member and press Enter to return to the File - Import ASG-Bridge Definitions pop-up.

Or

Type a member name in the Member Name field.

6 Specify whether you want imported Bridge Definitions to overwrite Bridge Definitions with the same name in the AKR you are importing into.

7 Specify the name of the AKR you want to import the Bridge Definitions into if it is not the AKR specified in the Data Set Name field.

8 Press Enter to Import the Bridge Definitions. A message displays confirming the number of Bridge Definitions imported.

9 Press PF3 to return to the screen or pop-up where you accessed the File - Import ASG-Bridge Definitions pop-up.

9 Bridge Event Log

The Bridge Event Log records significant events. You can enable or disable the Bridge Event Log depending on the needs and the requirements of your site. You can browse, print, or archive this log.

The Bridge Event Log record is a comma-delimited record that contains this information about the event being recorded:

- Date
- Time
- User ID
- AKR name
- Bridge Name
- Brief description of the event

Enable the Bridge Event Log by specifying the dataset name of a suitable file in the Bridge-Log-DSN parameter. (See the *ASG-Bridge Installation Guide* for more information about specifying and initializing the Bridge Event Log dataset.) The Bridge Event Log is disabled if you omit this parameter or if Bridge cannot locate the specified dataset.

The Bridge Event Log dataset name is shown in the Bridge Event Log DSN field on the Options - Product Parameters pop-up. If logging is disabled, this field is blank. (See "Verifying User Options" on page 14.)

When you initiate a Bridge session, the Bridge-Log-DSN parameter is examined to determine whether a dataset name is specified. If a dataset name is identified, an attempt is made to locate the dataset. If the dataset is found, the session records all significant events to the log for the duration of the session. If the dataset cannot be found, event logging is not active for the session, even if the dataset becomes available during the session.

Information writes to the Bridge Event Log for the duration of every Bridge session, and you must exercise diligence when managing this log. For example, browsing or running a utility against this log can impact the Bridge sessions that are writing to the log. Also, if you change the Bridge-Log-DSN parameter it causes new Bridge sessions to use the new log, but it does not affect the sessions that are already using this log. For this reason, ASG recommends that you process the Bridge Event Log off-hours or when no active Bridge sessions are open.

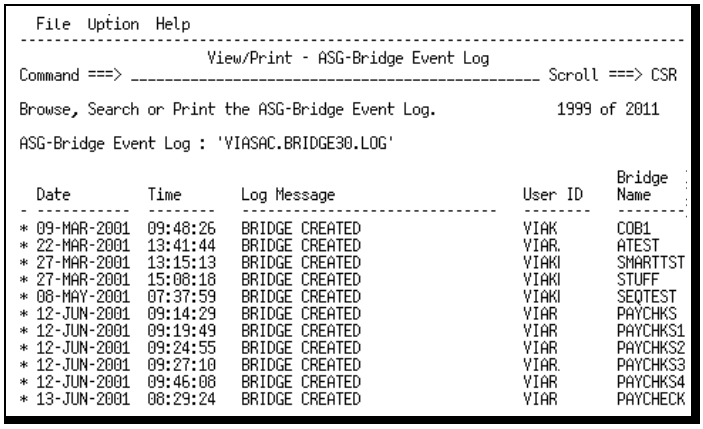
Note:

If the Bridge Log becomes full during a session, a message displays and logging is suspended.

To view the Bridge Event Log

1 Select View ▸ Event Log from the Bridge Primary screen and press Enter. The View/Print - ASG-Bridge Event Log screen displays (see Figure 141).

Figure 141 • View/Print - ASG-Bridge Event Log screen



```
  File  Uption  Help
--------------------------------------------------------------------------------
                      View/Print - ASG-Bridge Event Log
Command ===> _____________________________________________ Scroll ===> CSR

Browse, Search or Print the ASG-Bridge Event Log.              1999 of 2011

ASG-Bridge Event Log : 'VIASAC.BRIDGE30.LOG'

                                                                        Bridge
  Date         Time       Log Message                    User ID      Name
- -----------  --------   -----------------------------  --------     --------
* 09-MAR-2001  09:48:26   BRIDGE CREATED                 VIAK         COB1
* 22-MAR-2001  13:41:44   BRIDGE CREATED                 VIAR.        ATEST
* 27-MAR-2001  13:15:13   BRIDGE CREATED                 VIAKI        SMARTTST
* 27-MAR-2001  15:08:18   BRIDGE CREATED                 VIAKI        STUFF
* 08-MAY-2001  07:37:59   BRIDGE CREATED                 VIAKI        SEQTEST
* 12-JUN-2001  09:14:29   BRIDGE CREATED                 VIAR         PAYCHKS
* 12-JUN-2001  09:19:49   BRIDGE CREATED                 VIAR         PAYCHKS1
* 12-JUN-2001  09:24:55   BRIDGE CREATED                 VIAR         PAYCHKS2
* 12-JUN-2001  09:27:10   BRIDGE CREATED                 VIAR.        PAYCHKS3
* 12-JUN-2001  09:46:08   BRIDGE CREATED                 VIAR         PAYCHKS4
* 13-JUN-2001  08:29:24   BRIDGE CREATED                 VIAR         PAYCHECK
```

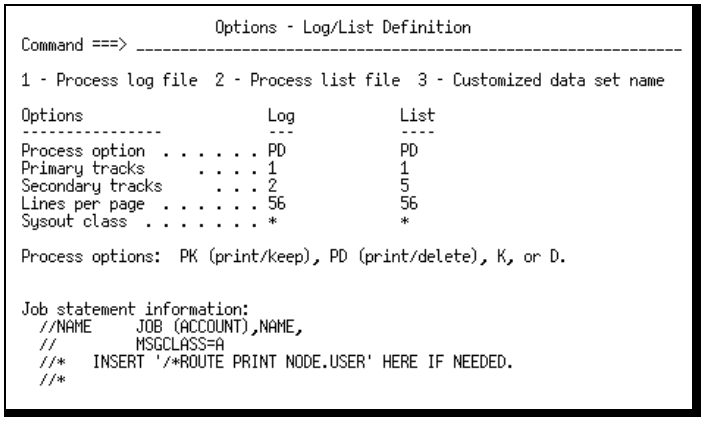
2 Press PF3 to return to the Bridge Primary screen.

To print the Bridge Event Log

1 Select File ▸ Print from the View/Print - ASG-Bridge Event Log screen and press Enter. A message displays specifying how many lines are printed to the List file.

2 Select Option ▸ Process list file to process the list file and press Enter. The Options - Log/List Definition pop-up displays (see Figure 142 on page 129).

Figure 142 • Options - Log/List Definition Pop-up



3 Specify options and press PF3 to process the list file and return to the View/Print - ASG-Bridge Event Log screen.

4 Press PF3 to return to the Bridge Primary screen.

10 Testing Your Bridge Routine

This chapter describes how to test your Bridge routines and contains these sections:

What is the VIAGTEST Program?

VIAGTEST is a program you can use to test Bridge Routines against your data without using any of your application programs. VIAGTEST reads your data and bridges all of the records with the Bridge Routine that you specify.

The program produces these two outputs:

- Output bridges to original format. This output should be an exact copy of the input file. (A file compare utility can be run against the input file and the old format, and every record should be an exact match. This verifies that the Bridge Routine you selected is defined correctly, and that it operates successfully on your data without inducing any error or unexpected change to it.)
- Output writes in new format. You can print, discard, or visually examine this output to ensure that it is bridged properly.

There are other uses of the VIAGTEST program. For instance, you can use it to generate a new format of the files that you input into the converted programs which process the file into its new format. Or, you can use VIAGTEST as a simple file conversion program.

Using the VIAGTEST Program

To use VIAGTEST

1. Make a copy of the VIAGTEST JCL member of the CNTL library.
2. Verify that the job card contains valid accounting information and other parameters applicable to your needs.
3. Change the EXEC statement PARM field to specify the name of the Bridge Routine you want to test.
4. Add a concatenation to the STEPLIB, specifying the dataset name of the Generate Library containing the Bridge Routine.
5. Change the IN DD statement, specifying the dataset name of the file containing the data you want used in the test.
6. Change the OUTOLD DD statement to specify the dsname of the file you want to create to contain the old format records. This file should be a mirror-image of the input file. Be sure to specify the DCB RECFM, LRECL, and BLKSIZE, as this information is not available from the VIAGTEST program. Make sure the LRECL should be the same as that of the IN file being processed.

Note:

If you are reading VSAM datasets, the OUTNEW and OUTOLD datasets must be sequential or SYSOUT datasets. The dataset on DDNAME IN can be sequential, SYSIN, or VSAM.

7 Change the OUTNEW DD statement to specify the dsname of the file you want to create that will contain the new format records. Again, be sure to specify the DCB RECFM, LRECL, and BLKSIZE. Also, remember that the LRECL and BLKSIZE are different than that specified on the OUTOLD statement, because records being written to this file contain expanded fields and are longer.

Note:

As a general rule, you can compute new LRECL as the original LRECL plus the number of bytes you are adding for each data field being bridged. For example, assume the original LRECL is 123 bytes. If the file contains records with six fields being bridged and each field expanding by two bytes, the record is expanding by 12 bytes. The new LRECL is 123+12, or 135 bytes.

8 Optionally, you may add the VIAGTRAC DD statement if you are having difficulty with your Bridge Routine and want to internally trace the bridging process.

Run the job and examine the output, including the messages generated by Bridge. The job should end with a zero condition code.

Bridging Direction Parameter

This parameter specifies whether you want to forward or reverse bridge the input file. Parameter values allowed are FORWARD and REVERSE. If you do not specify a direction parameter, the value FORWARD is assumed. You may specify just the first letter of the parameter values (R or F).

Examples

This is the syntax that initiates forward bridging and allows the direction parameter to default:

```
//RUNTEST EXEC PGM=VIAGTEST,PARM='VIAGBDM'
```

This is the syntax that initiates forward bridging and explicitly specifies the direction:

```
//RUNTEST EXEC PGM=VIAGTEST,PARM='VIAGBDM,FORWARD'
```

This is the syntax that initiates reverse bridging:

```
//RUNTEST EXEC PGM=VIAGTEST,PARM='VIAGBDM,REVERSE'
```

Input File Start and Stop Options

You specify input file start and stop options on control statements that you supply through a SYSIN file. These options give you the flexibility to start and stop testing at a given record number.

STARTAT Parameter

The STARTAT option instructs VIAGTEST to position the input file at a specific point before beginning the bridging test. The parameter value is a number representing the `nth` record where processing will begin. For example, to start processing at record 1000, code `STARTAT=1000` in the input statement. If an insufficient number of records exist in the file to complete the STARTAT processing, VIAGTEST terminates with a return code to indicate insufficient records. For example:

```
STARTAT=nn
```

STOPAFT Parameter

The STOPAFT option instructs VIAGTEST to terminate after processing a specified number of records. The parameter value is a number representing the number of records to process. For example, to terminate processing after bridging 500 records, code `STOPAFT=500`. If an insufficient number of records exist in the file to complete the request, VIAGTEST terminates with a return code to indicate insufficient records. For example:

```
STOPAFT=nnn
```

Using STARTAT and STOPAFT Parameters

You must specify the STARTAT and STOPAFT options individually on separate control statements. If you code an option more than once, Bridge uses the last value it encounters. If an option is incorrectly specified or unrecognized, VIAGTEST displays a descriptive error message and terminates with a return code, for example:

```
//STEPLIB  DD
//IN       DD DSN=MY.ORIGINAL.DATA,DISH=SHR
//OUTNEW   DD DSN=NEW.BRIDGED.DATA,DISP=SHR
//OUTOLD   DD DSN=OLD.BRIDGE.DATA,DISP=SHR
//SYSIN    DD *
STARTAT=1000
STOPAFT=5000
/*
.
```

Return Codes

These are the return codes, causes, and recommended actions:

Return Code	Cause and Recommended Action
0	VIAGTEST completed successfully.
4	VIAGTEST completed successfully at end of file, but a STOPAFT option was not completed.
8	VIAGTEST did not complete successfully. There are numerous causes of this return code. A descriptive error message displays in the message log describing the cause of the problem.

Messages from VIAGTEST

VIAGTEST displays these messages on the message log:

`VIAG0901-I TOTAL RECORDS INPUT: nnnnnnnnn`
Indicates the number of records VIAGTEST found in the input file.

`VIAG0902-I STARTING AT RECORD: nnnnnnnnn`
Indicates the starting position in the input file, which is normally 1. However, if you specified a starting position with a STARTAT= parameter, this message reflects the specified record number.

`VIAG0903-I STOPPING AFTER RECORD: nnnnnnnnn`
Indicates the stopping point in the input file when the STOPAFT= parameter is used. The value displayed is the actual record number stopped at after the starting position is determined. For example, if the starting position is the first record, the stopping position is the same as the STOPAFT= value. However, if the starting position is record 100, and the STOPAFT= value is 200, the actual stopping point is 299. That is, record 100 is the first record processed, and 299 is the last - a total of 200 records.

`VIAG0904-I TOTAL RECORDS WRITTEN: nnnnnnnnn`
Indicates the number of records written to each output file. If you do not specify STARTAT and STOPAFT parameters, this count is the same value as the value displayed in message VIAG0901-I. If you specify a STOPAFT parameter the value displayed is the lesser of the STOPAFT count, or the records remaining to process if a STARTAT parameter is specified.

Troubleshooting VIAGTEST

If the job abends and you provided the correct information in the EXEC PARM and DD statements, examine the message log for Bridge messages (they all begin with `VIAG`) to determine the cause of the error. Usually, the problem is caused by an incorrectly defined Bridge Routine, or the Bridge Run Time facility found either an error or inconsistency between the Bridge Routine and the data being processed.

If you are unable to determine the cause of the problem, contact your Bridge administrator or the ASG Service Desk for further assistance.

Testing IMS Bridge Definitions

These are the test utilities Bridge provides for testing IMS Bridge Definitions:

- VIAGFDLT, used for forward bridging
- VIAGRDLT, used for reverse bridging

These test utilities begin bridging and then invoke the standard IMS utility DFSDDLT0. You can use any customized JCL and parameters for DFSDDLT0 by changing the program name. This also enables you to test for any DL/1 calls.

The sample JCL member, VIAGJDLT, is available in the CNTL library. This sample prints the contents of all segments in the IVP database after bridging.

Note:

For more information about the DFSDDLT0 IMS utility, see the IMS Application Programming Guide: DB for IMS Version 5 and the IMS Application Programming Guide: DL/1 Calls for IMS 4.1.

11 Program Integration

This chapter describes how to integrate Bridge into application programs and contains these sections:

To integrate Bridge into your program, simply add a single CALL statement to the program. This call invokes the Bridge initialization module which, in turn, initializes Bridge, loads, and prepares the required Bridge Routines for execution.

A different Bridge Initialization module exists for each supported programming language or environment. It is important you specify the correct module name in the CALL statement. Some Bridge initialization modules require one or more parameters. See the appropriate section for more information.

These are the supported environments and the corresponding module name for each:

Module Name	Environment/Language Description
VIAGFCOB	All supported COBOL language compilers (See "Integrating Bridge into Batch QSAM/VSAM Programs" on page 139)
VIAGFPLI	All supported PL/I compilers (See "Integrating Bridge into Batch QSAM/VSAM Programs" on page 139)
VIAGFASM	Assembler language (See "Integrating Bridge into Batch QSAM/VSAM Programs" on page 139)
VIAGFCIC	CICS programs, regardless of language (See "Integrating Bridge into CICS Programs" on page 145)
VIAGFIMS	IMS programs, regardless of language (See "Integrating Bridge into IMS Programs" on page 148)
VIAGFDLI	CICS DL/I programs, regardless of language (See "Integrating Bridge into CICS Programs" on page 145)
VIAGFIDM	IDMS, COBOL batch programs (See "Integrating Bridge into IDMS Programs" on page 163)
VIAGFIDC	IDMS, online programs (See "Integrating Bridge into IDMS Programs" on page 163)

Note:

These module names reflect forward bridging. To perform reverse bridging, replace the `F` in the fifth position of the module name with `R`. For example, use the module name VIAGRCIC to invoke reverse bridging in a CICS program.

Integrating Bridge into Batch QSAM/VSAM Programs

Parameters pass to Bridge in one or more pairs. For each parameter pair, the first parameter identifies the DD name of the file Bridge processes. The second parameter specifies the name of the file's respective Bridge Routine Name.

These are the rules for coding Bridge initialization statements:

- The DD name must be an eight-byte field containing the DD name of the file you want bridged, left-justified, and right-filled with spaces as necessary.
- The Bridge Routine name must be an eight-byte field containing the name of the Bridge Routine, left-justified, and right-filled with spaces as necessary.
- The CALL to Bridge must be executed before any of the affected files are opened by your program.
- The file must be opened by your program to activate bridging. If the file is opened by another program, bridging is not activated.
- The Bridge Routine must reside in a Generate Library that is in a link list library, a STEPLIB, or a JOBLIB.

Note:

Triggering Bridge for VSAM and sequential files affects all programs whether statically link-edited or dynamically loaded.

This is the syntax of this CALL for COBOL:

```
CALL VIAGFCOB USING DD name-1,  bridge-routine-name-1,
                    DD name-2,  bridge-routine-name-2, ....
```

This is a typical COBOL CALL to the Bridge initializing routine:

```
CALL VIAGFCOB USING EMPIN, EMPDATES,
           EMPOUT, EMPDATES,
           DEVDEFS, DEFDATES
```

In this example, three parameter pairs are passed to VIAGFCOB, the Bridge COBOL interface module. Each parameter pair consists of the DD name of the file Bridge will process and the name of its respective Bridge Routine.

The syntax of this CALL for PL/1 is:

```
DCL VIAGFPLI ENTRY OPTIONS(ASM, INTER, RETCODE);
CALL VIAGFPLI(DD name-1,  bridge-routine-name-1,
DD name-2,  bridge-routine-name-2, ...);
```

The syntax of this CALL for Assembler is:

```
CALL VIAGFASM,(DD name-1,bridge-routine-name-1,
 DD name-2,bridge-routine-name-2,...),VL
```

Restrictions

COBOL Restrictions

- Issue the call to VIAGFCOB from the program that issues the open statement for that file.
- For an alternate index, specify the file and the alternate index file in the VIAGFCOB CALL.

PL/I Restrictions

- Link the VIAGFPLI routine with the program

  Note:
  FETCH and RELEASE should not be used for this routine.

- Do not define the record descriptions specified in the Bridge Definition as BASED or CONTROLLED
- Give a VSAM file with a variable record length a record length large enough to hold the largest converted record

These commands and statements are not supported:

- PUT statements
- SCALARVARYING
- READ BACKWARDS
- EVENT
- VSAM alternate index

Supported Environments

These programming languages are supported:

- PL/1 versions 1.5, 2.3, and 370.
- COBOL II
- COBOL 370
- COBOL for MVS and VM
- COBOL for OS/390, no support for object oriented components

These file types are supported:

- QSAM files
- VSAM files

Controlling Bridging at the File Level

Bridge facilitates a program-centric strategy of converting system files and programs. This means that you can convert programs and put them into production one at a time or in groups or batches while continuing to process unconverted files. Use this strategy to postpone file conversions.

After you have converted of all of your programs, you might think about converting your files. Suddenly, many questions come to mind. How do I convert my files? One at a time? All at once? How does this affect bridging, and how and when do you turn it off?

Although Bridge is integrated into each of your programs, you actually control bridging at the file level. You can turn bridging on and off for each individual file without making any program or JCL changes. One simple action in the Bridge interactive facility immediately disables all bridging of a particular file in all programs without having to change any of the programs. You do not need to change any programs now or at any time in the future to disable bridging.

You can schedule file conversions one at a time or in groups, and disable bridging for each file as conversion completes. After conversions are finished, you can remove the CALLs to Bridge. This can be done during normal maintenance or as time permits.

After you convert a file, you need to disable bridging. To disable bridging, you simply indicate that a particular file is converted. Access the Bridge Definition - Attributes screen for the Bridge Definition that is related to a particular file and set the File Conversion Complete indicator to `YES` and regenerate the executable Bridge Routine. Thereafter, as a program tries to initialize bridging for this file, the Bridge Routine indicates that the file is converted and no longer requires bridging, and bridging is automatically disabled. Bridging of other files in the same program is unaffected. These are the affects of the File Converted Status on Forward and Reverse Bridging:

File Converted	Forward Bridging	Reverse Bridging
NO	Enabled	Disabled
YES	Disabled	Enabled

Reverse bridging for an unconverted file is automatically disabled for that file. Conversely, if the File Converted status is `YES`, Reverse Bridging is allowed. This is why bridging is file-centric: After integrating Bridge into all of your programs, you control bridging at the file level.

Note:

The above discussion assumes that you defined just one Bridge Definition for each file, and that by changing the Bridge Definition and regenerating the Bridge Routine, you effect bridging for every file being bridged with that Bridge Routine. If you defined more than one Bridge Definition for a particular file, then this situation becomes more complex, although the principles remain the same.

Overriding Bridge Routines

To override parameters specified in the Bridge Definition, add the VIAGRTO DD statement to the Bridge program's member. This DD statement should specify a sequential fixed 80-byte-length file and may contain overriding parameters. This is the syntax for records in this file:

This statement overrides the FILE-CONVERTED status indicator in the Bridge Definition:

```
DD name, FILE-CONVERTED={ YES | NO }
```

This syntax specifies that the file is not converted. Bridging should take place for the specified DD name:

```
FILE-CONVERTED=NO
```

This syntax specifies that the file is converted. Bridging should not take place for the specified DD name:

```
FILE-CONVERTED=YES
```

The DD name you specify must have a generated Bridge Routine and the program must invoke the Bridge Initialization Routine.

After a data file is converted, change the file conversion status indicator in the Bridge Definition from `NO` to `YES`, and regenerate the Bridge Routine. This disables bridging for that file even though Bridge is still called by your program. The Bridge Routine indicates that the file is converted, and disables bridging for that file. Bridging of other files in the same program is unaffected.

However, if a program can access files that are not converted (for example backup or historical files) you can specify the overriding parameter `FILE-CONVERTED=NO` in the program's JCL. This causes Bridge to convert records from the historical file. In this way, a converted program that usually accesses converted files can also access non-converted historical files.

Run-Time Debugging

The Bridge run-time component contains a built-in trace facility that logs detail information about the status and activity of the bridging process, including before and after images of records being bridged. This trace log helps you diagnose problems with a particular bridging operation.

To create this log, simply add this DD statement to your program's JCL:

```
//VIAGTRAC DD SYSOUT=*
```

where '*' is your preferred SYSOUT output class.

This DD statement triggers the creation of the trace log. Be aware that this log is highly detailed and Bridge may generate a large amount of trace output, depending on the number and size of the records being bridged. After your diagnosis, remove the VIAGTRAC DD statement from the JCL.

Controlling Bridge Trace Log Content

Optionally, you can control the content of the trace log to a subset of the normal output. This is accomplished by including a VIAGTCTL DD statement in your JCL that is followed by or points to either a dataset or PDS member containing one or more trace control statements.

This is the format of the Trace Control Statement:

```
TRACE=parm1,parm2
```

where:

parm1 identifies the module or modules being traced. You can specify:

- ALL - messages are printed from all modules.
- Two-letter module type - messages are printed from the module types indicated. These are the valid module types:

Module Type	Description
GB	Batch Interface modules
GC	CICS Interface modules
GG	Bridge Routine and PCT Generation modules
GI	IMS Interface modules
GS	Common or System modules

- Eight-letter module name - messages are only printed from the module specified.

parm2 is a two-character hexadecimal code that determines the type of trace message you want. This two-character code represents a byte of bit-flags. Each bit flag described in this table represents a specific trace message type:

Flag	Description
80	Trace ON
40	Trace TYPE=ENTER messages only
20	Trace TYPE=EXIT messages only
10	Trace IO messages only
00	Trace OFF

Bit-flags can be combined to select more than one type of message at a time. For example, the code 60 indicates that you want both TYPE=ENTER and TYPE=EXIT trace messages. If parm2 is omitted, all message types are traced.

Note:

Bridge Trace Log message printing can place heavy demands on a system, substantially affecting execution time and available spool capacity. Therefore, use this facility with caution when attempting to diagnose a run-time problem. Always be sure to remove the VIAGTRAC DD statement from you JCL when you complete your testing.

Integrating Bridge into CICS Programs

Bridge supports CICS programs that access FCT-defined VSAM files. The records are either fixed length format or variable length format.

Note:

If you bridge fields within VSAM keys in remote files, a new program (VIAGCRFI) executes in the remote file-owning region. A CICS definition must be made in the remote file-owning region and the Loadlib where VIAGCRFI resides must be in the DFHRPL for that region. No transaction definitions are necessary.

You must add a CALL statement to the program that will call the Bridge Initialization Module. The routine name is the same for all supported languages, VIAGFCIC. As with the batch program calls, parameters pass to Bridge in pairs and specify the file's FCT and SYSID name and the name of the respective Bridge Routine.

Caution! The Bridge CICS initialization module issues several command-level CICS requests during its initialization processing. The EIB is modified by these requests during their execution. If your program saves EIB contents for later reference, insert the Bridge CALL statement after the code that saves the EIB contents.

The syntax of this CALL for COBOL is:

```
CALL VIAGFCIC USINGfile-id-1,  bridge-routine-name-1,
         file-id-2,  bridge-routine-name-2, ....
```

The syntax of this CALL for PL/1 is:

```
DCL VIAGFCIC ENTRY OPTIONS(ASSEM, INTER, RETCODE);
CALL VIAGFCIC(file-id-1,  bridge-routine-name-1,
file-id-2,  bridge-routine-name-2, ...);
```

The syntax of this CALL for ASSEMBLER is:

```
CALL VIAGFCIC,(file-id-1,bridge-routine-name-1,file-id-2,
bridge-routine-name-2, ...),VL
```

The file-id identifies the file referred to by an EXEC CICS command. The Bridge CICS interface requires that you specify the FCT entry name and SYSID of the file bridged, unlike the batch program calls where the file is identified by its DD name.

The file-id is 12 bytes long and consists of an 8-byte FCT entry name and a 4-byte SYSID. If the file is a local file, or is defined in the FCT, the SYSID part of the file-id contains blanks. Otherwise, the SYSID part of the file-id contains the SYSID of the system where the file resides. This means that if the file-id specified with non-blank SYSID, EXEC CICS commands specified without a SYSID parameter are not intercepted, although the specified file name may be directed in the FCT to the same SYSID.

Bridging occurs only when a file is accessed by the program calling the VIAGFCIC module, which passes the file's file-id and Bridge Routine name. Bridging does not occur when that file is accessed by a program that did not call the VIAGFCIC module, even if it calls or is called by a program that called VIAGFCIC, for example:

PROGA (bridge not active for program)

```
   EXEC CICS LINK PROGRAM('PROGB')
```

PROGB (bridge active for program)

```
   CALL VIAGFCIC ........
   EXEC CICS LINK PROGRAM('PROGC')
```

PROGC (bridge not active for program)

```
   EXEC CICS LINK PROGRAM('PROGD')
```

PROGD (bridge active for program)

```
   CALL VIAGFCIC.......
```

CICS ABEND Codes

In some cases errors caused by invalid Bridge Routine names or environmental problems occur. In these cases, the transaction program abends with one of these codes:

- B2K0 - Bridge is not activated in the CICS system.
- B2KU - An application error. Usually bad parameters were passed to VIAGFCIC.
- B2KD - A Bridge internal error was detected.

Bridge abend codes are accompanied with a detailed message in the JOBLOG for the CICS region.

System Notes

Supported Environments

CICS 3.3, 4.1, 5.1, and 5.2 are supported.

Automatic Initialization of Bridge Fails

Consider Bridge a CICS extension and keep it active. Specify Bridge in the PLT.

If you initialize Bridge from the PLT and it fails, a message is sent to the system operator (WTO), with a request to either allow CICS to start without Bridge (reply `GO`), or not to start (reply `CANCEL`).

- A `GO` reply starts CICS without initializing Bridge. However, this may result in B2K0 transaction abends if a program requests bridging.
- A `CANCEL` reply terminates the CICS system.

Initializing Bridge Manually

You can manually activate Bridge by entering the transaction `B2KD,ON` at a CICS terminal. If the system fails to start, a message is issued.

De-activating Bridge in the CICS System

Normally, Bridge deactivates automatically when the CICS system shuts down. However, you can manually deactivate Bridge if necessary. To manually deactivate Bridge, enter the transaction `B2KD,OFF` at a CICS terminal.

Note:

Use manual deactivation with caution. B2K0 transaction abends may occur if a program calls Bridge when Bridge is deactivated.

User Exit Routine

If required, a user exit can be called for each file specified in a VIAGFCIC call. Specify the user exit routine in the Bridge run-time parameters. The exit routine is a standard CICS program that receives control from an EXEC CICS LINK command. The system links to the user exit routine with the data specified in the VIAGEPRM copybook.

Defining Bridge Routines to CICS

You can define all Bridge Routines to CICS in the PPT by using CEDA or in AUTO INSTALL. All Bridge Routine Generate Libraries should be in the DFHRPL concatenation. If you are accessing remote files, you must define the program VIAGCRFI in the file-owning region and make it available to that region through an entry in the DFHRPL concatenation.

Restrictions

A program may pass up to 50 parameter pairs to Bridge CICS Interface Module.

Integrating Bridge into IMS Programs

Bridge provides forward and reverse bridging support for BMP, DLI, and MPP programs. These programs use CBLTDLI, PLITDLI, and ASMTDLI interfaces to access IMS.

Bridge supports IMS versions 3.1, 4.1, 5.1, and 6.1.

Definition of Fields

You define Bridge Definitions for each DBD containing segments with fields that you want converted. Bridge generates Bridge Routines from these definitions for each DBD. These routines convert data during IMS calls. One Bridge Definition is created for each DBD that Bridge will handle.

Bridge uses COBOL or PL/I fields that are members of a structure that maps a segment. These fields may or may not be defined as IMS fields. To simplify the definition process and to avoid errors, Bridge uses the Bridge Definition in conjunction with the original DBD, and uses the new DBD that contains the fields in a new format to construct DBD Conversion Routines (DCRs).

In a similar manner, Bridge uses the contents of the PSB and its related DCRs to generate PSB Conversion Routines (PCRs).

IMS Support Overview

A converted IMS program triggers Bridge from a single call added to a Bridge initialization routine. This routine activates the IMS run-time component of Bridge. After activation, Bridge intercepts all IMS calls, handles the IMS commands related to converted DBDs, and uses the appropriate Bridge Routines to handle these commands:

- Retrieve command (GU, GN, etc.)

  Bridge converts data from the old format to the new format and presents the converted data to the program.

- Update command (ISRT, REPL, DLET)

  Bridge converts data from the new format to the old format, issuing the update command with the converted data.

For each IMS command, Bridge handles data conversion in the I/O area, the data fields in the SSA, and the key feedback area of the PCB mask.

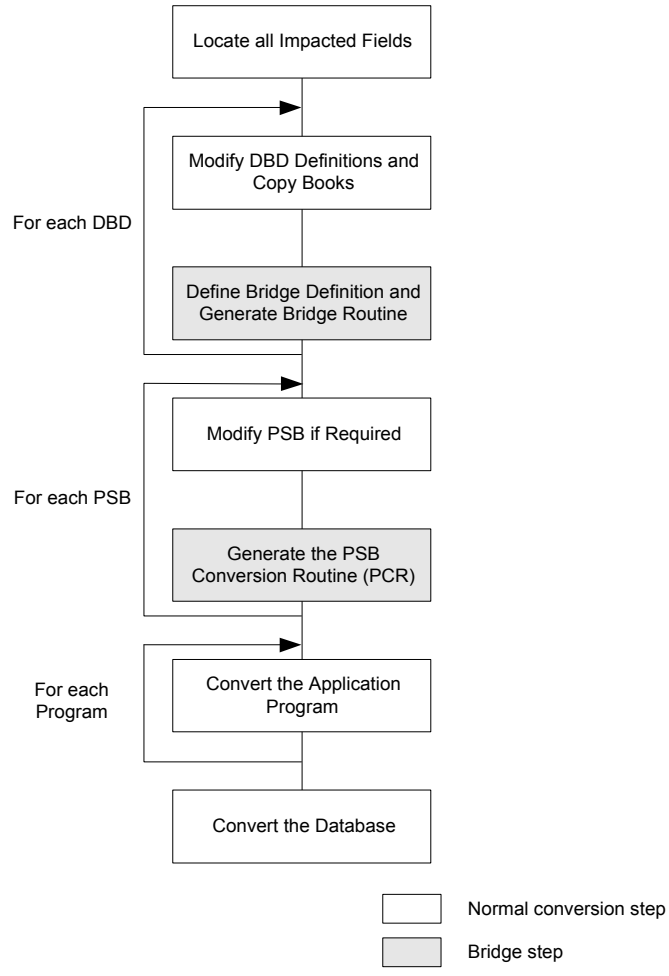
By using Bridge, a converted IMS program can access an unconverted database as if it were converted. When the database is finally converted, the Bridge Routine can be disabled and the application program continues to run without recompiling or further modification.

Bridge initializes at the program level so you can simultaneously access the same DBD from either a converted or an unconverted program. Each program receives and processes the data in the applicable format.

The Conversion Process - IMS

Converting programs and databases for projects, such as euro currency compliance, includes several steps performed at the IMS database and program levels. Incorporating Bridge into a conversion project influences several steps in the process. Figure 143 outlines the usual steps of a normal conversion and includes the additional steps to implement Bridge in the process.

Figure 143 • IMS Conversion Process Using Bridge



Conversion Description

To convert programs and databases

1. Identify all fields in the database you want to convert. This step is a direct outcome of the impact analysis.

2. Modify the DBD to reflect the new field format. Modify the copybooks accordingly. The modified DBD is stored in a load library that does not affect the operating IMS. Bridge uses these DBDs; IMS can use them after the database is physically converted.

3. Define a Bridge Definition for each DBD with the Interactive Facility of Bridge and generate the executable Bridge Routines. You can use these routines to convert the DBD fields. A logical DBD does not require a Bridge Definition. The Bridge Routines of the physical DBDs (from which its segments are derived) generates the Bridge Routine.

4. Some changes in field formats may require you to update positions in PCBs that specify these fields as SENFLDs. Modify all PSBs that contain such PCBs.

5. For each PSB, generate the PSB-level Conversion Routine (PCR) by running a Bridge batch job. No other input from you is required in this step.

6. Convert the application program. Add a call to the program's initialization routine to trigger Bridge. Unit test the converted program, and put it back into production.

7. After all programs are converted, you can physically convert the database and disable Bridge.

Defining DBD Conversion Routines

Bridge converts fields so that they display in a specified new format. To accomplish this conversion, Bridge needs information on all database fields. You define the fields to Bridge in a Bridge Definition.

Bridge Definition

For each physical DBD Bridge handles, create a Bridge Definition that defines all of the segments containing conversion fields in the DBD. Bridge processes the Bridge Definition to generate an executable Bridge Routine that contains all the information and logic needed for the conversion of segments in the defined DBD.

A logical DBD does not require a Bridge Definition. Bridge gathers the information from the Bridge Routines of the physical DBDs it references. The Bridge Routine loads into memory at run-time and is invoked either to convert data or to get information about the DBD.

Bridge requires a Bridge Routine for a logical DBD that references a physical DBD that contains fields that can convert, even if the logical DBD itself contains no such fields. Furthermore, all physical DBDs referred to by such a logical DBD must have corresponding Bridge Definitions and Bridge Routines. You must generate Bridge Routines for all physical DBDs referred to by a logical DBD before you generate the Bridge Routine for the logical DBD (see "Generating a DCR for a Logical DBD" on page 153).

The Bridge Definition is similar to those of Sequential and VSAM files. Most importantly, you must associate segment names with data structures in the Bridge Definition for Non-GSAM DBDs. The conversion fields in each segment or GSAM record are defined in fields in a COBOL or PL/I structure that maps the defined segments/GSAM record. A copybook (INCLUDE file) maps each segment/GSAM record.

These are the components of a Bridge Definition that depend on the IMS environment type indicated in the Bridge Definition:

Environment Type	Bridge Definition Description
Physical Non-GSAM DBD	Each segment containing one or more conversion fields is defined, and each conversion field is assigned a Bridge Rule
Logical DBD	No Bridge Definition is required All the needed information is derived from the Bridge Routines of the physical DBDs it references
GSAM DBD	No segments are defined Conversion field definition is similar to Sequential/VSAM files

When you define a non-GSAM Bridge Definition, it is important to associate a segment name with the corresponding segment structure (see "Defining IMS Segment/Record Association" on page 64). This is necessary so that Bridge can associate the segment name with a data structure and map the conversion fields in the data structure to fields in the segment. Bridge assumes that the relative positions of the fields in the data structure correspond to the relative positions of the conversion fields in the IMS segment, and that all fields are the same except those that have Bridge Rules assigned to them.

The Bridge Routine uses calculated positions of the conversion fields to convert segments that the application program accesses as a whole (no SENFLD in the PCB definition). The Bridge Routine also uses these positions to identify IMS fields that contain conversion fields. The Bridge Routine uses field level conversion routines to convert fields in a segment view (SENFLD in the PCB definition), used fields in the SSA, and for the concatenated key.

Checks During the Bridge Routine Generation

These are the checks performed by the Bridge Routine Generator during Bridge Routine generation:

- Each conversion field is fully contained in an IMS field. If a specified conversion field is not fully contained in an IMS field (for overlapping IMS fields - in the relevant IMS fields), a warning message is issued.

  If an application program accesses such a field during run-time, either in the SSA or the I/O area, an error message is issued and the program is abnormally terminated. This occurs because Bridge cannot determine how to convert the partially-specified field.

  There are cases where conversion field definitions are logical and valid, and therefore during generation a warning is issued, and not an error-level message. Converting a segment that is defined in the PSB without SENFLDs is valid, even if it contains conversion fields that overlap IMS fields defined in the DBD.
- All segments in the old DBD exist in the new DBD.
- All fields in the old DBD (and only those fields) exist in the new converted DBD, are defined in the same order, and are the same size. A field can have a different size only if the difference corresponds exactly to the length difference derived from the Bridge Rules assigned in the Bridge Routine.
- Each segment defined in the Bridge Definition should exist in the DBD.
- For a logical DBD:
  - There is a Bridge Routine for each physical DBD referred to by the logical DBD. The Bridge Routines of the physical DBDs must be generated before the Bridge Routine for the logical DBD is generated.
  - All the Bridge Definitions of the physical DBDs referred to by a logical DBD are defined with the same File Converted status.
- If generating a Bridge Routine for a DBD with a logical child, a Bridge Routine exists for the DBD where the paired real logical child segment is defined.

Generating the Bridge Routine

After you define the Bridge Definition, you can generate the Bridge Routine. Bridge scans the original DBD and the new converted DBD and determines the IMS fields in the new DBD that correspond to those in the old DBD.

The generation process creates a load module called a Bridge Routine that contains routines and information required to efficiently convert data.

This information comprises the generated Bridge Routine:

- I/O conversion routines to convert entire segments
- Field conversion routines to convert specific fields
- Information on all segments in the DBD
- Information on all fields in each segment that needs conversion

You must regenerate a Bridge Routine in these situations:

- The DBD is modified.
- For a DBD with a virtual logical child, whenever the Bridge Routine of the DBD containing the paired real logical child is modified and its Bridge Routine is regenerated.
- For a logical DBD, when one of the physical DBDs referred to by the logical DBD is modified and its Bridge Routine is regenerated

Generating a DCR for a Logical DBD

The procedure that creates the DCR is called the DCRGEN. To generate a DCR for a logical DBD, submit a job that executes the cataloged procedure VIAGJGLD. Specify these symbolic parameters:

Parameter	Specify
DCR=dcrname	Name of the DCR to generate
OLDDBD=*libraryname*	Dsname of the old DBD load library
DCRLIB=*libraryname*	Dsname of the DCR library that stores this DCR

Example:

```
//DCRGEN JOB (jobacct), ),.....
//S1 EXEC VIAGJGLD,DCR=MYDCR,
//    OLDDBD='MY.OLD.DBD.LOADLIB',
//    DCRLIB='MY.DCR.LOADLIB'
//
```

Note:

Sample JCL to execute the cataloged procedure VIAGJGLD is available in the CNTL library member VIAGJGDC.

Gathering Information on the PSB

The Bridge Routine provides most of the information Bridge needs. However, some additional information necessary for the conversion process is defined in the PSB. Bridge gathers this information from the PSB and stores it in a PSB Conversion Routine (PCR).

The PSB Conversion Routine

The procedure that creates the PCR is called PCRGEN. PCRGEN collects the necessary information and builds a load module to contain the information. Bridge loads this module into memory and uses it in the conversion process.

This is the input for the PCRGEN needs this input:

- DCRs for the DBDs that need converted and that are included in the PSB.
- The PSB:
  - If the old PSB is valid for the new converted DBD, the old generated PSB may be used.
  - If the old PSB is not valid for the new converted DBD and changes are made to the PSB to comply with the new field lengths, a new PSB should be generated. The new generated PSB is stored in a load library not affecting the operating IMS system. The PCRGEN uses the newly generated PSBs. The new PSB can be used by IMS when all DBDs referenced in it are physically converted.
  - The key length of the old PSB must be large enough to contain the largest concatenated key of the converted database because the key length defined in the PSB is used by IMS to allocate the area for the key feedback area in the PCB-mask. When activating a program, IMS uses the old PSB. Bridge converts the concatenated key into the area allocated by IMS. Therefore, the key length must be large enough to contain the largest converted concatenated key.

PCR Generation

PCRGEN is required for the PSB referenced by a converted program. Whenever a PSB is generated, run the PCRGEN. ASG recommends that you bind PCRGEN to PSBGEN.

When you create a new Bridge Routine for a PSB DBDs, PCRGENS are required. You must generate the PCR after you generate all relevant Bridge Routines.

Generating a PCR

To generate a PCR, submit a job that executes the cataloged procedure VIAGJGIP. Specify these symbolic parameters:

Parameter	Specify
PSB=psbname	Name of the PSB to process
OLDPSB=*libraryname*	Dsname of the old PSB load library
NEWPSB=*libraryname*	Dsname of new PSB load library. **Note:** If the old PSB name is valid for the new converted DBD, specify the same load library name for the NEWPSB=*libraryname* parameter that you specified for OLDPSB=*libraryname* parameter
DCRLIB=*libraryname*	Dsname of DCR library containing DCRs that correspond to DBDs referenced by this PSB
PCRLIB=*libraryname*	Dsname of PCR library that stores this PCR

For example:

```
//PCRGEN JOB (jobacct),.....
 //S1 EXEC VIAGJGIP,PSB=MYPSB,
 //   OLDPSB='MY.OLD.PSB.LOADLIB',
 //   NEWPSB='MY.CONVERTD.PSB.LOADLIB',
 //   DCRLIB='MY.DCR.LOADLIB',
 //   PCRLIB='MY.PCR.LOADLIB'
 //
```

Note:

Sample JCL to execute the cataloged procedure VIAGJGIP is available in the CNTL library member VIAGJGPC.

IMS Program Integration

A converted IMS program triggers Bridge by calling the initialization routine VIAGFIMS.

After Bridge is initialized, it intercepts all IMS calls and handles IMS commands such as GU, GN, and the ISRT, REPL, and DLET commands.

For each IMS command, Bridge handles the conversion of the data in the I/O area, the date fields in the SSA, and the key feedback area of the PCB-mask. Changes made by an application program to the PCB-mask cause unpredictable results and are restricted by IMS. Bridge assumes that this restriction is followed. Changes made by an application program to the PCB-mask might be overridden by Bridge.

Invoking Bridge in an IMS Program

Add a CALL to one of these Bridge initialization programs at the beginning of the program before an IMS call is executed:

Module Name	Bridging Requirements
VIAGFIMS	A converted program that accesses an unconverted database
VIAGRIMS	A unconverted program that accesses a converted database

There are no operands or parameters to pass. After this call is executed, Bridge intercepts and converts all relevant IMS data accessed by the program as well as any other program to which it passes control.

Linkage Considerations

The Bridge initialization routine may be statically linked with the application program or loaded dynamically.

The batch and IMS interface modules and intercepts are re-entrant to allow customer programs to call a Bridge Interface Module loaded into and executed from protected storage.

Combining IMS and Sequential/VSAM Bridging

The Bridge Sequential/VSAM interface and IMS interface coexist yet operate independently. If you want to activate both interfaces, include calls to each respective Bridge initialization routine in the application program. You can activate either Bridge interface first. Activating one does not affect the other, regardless of whether it is active.

Note:

Triggering Bridge for VSAM and Sequential files affects all programs whether statically link-edited or dynamically loaded. Similarly, triggering Bridge for IMS affects all programs that run under the same TCB as the program that triggered Bridge.

Restriction

You cannot combine converted and unconverted clusters; converted IMS programs should not invoke non-converted IMS programs, and non-converted IMS programs should not invoke converted IMS programs.

Run-time Issues

DCR and PCR Libraries

The Bridge IMS and DL/I Support Options construct and use two types of executable Bridge Routines: DBD Conversion Routines (DCRs) and PSB Conversion Routines (PCRs). A DCR contains bridging routines and information about the DBD that it converts. A PCR contains bridging routines and information about the PSB that it converts.

DCRs are named the same as the DBDs that they convert, and PCRs are named the same as the PSBs that they convert. Consequently, DCRs and PCRs must be treated differently than DBDs and PSBs to prevent conflict when trying to load one or the other from their respective libraries. Bridge uses special DCR Libraries and PCR Libraries.

Specifying the IMS and DL/I Support DCR and PCR Libraries

For IMS support, Bridge loads DCRs and PCRs from their respective libraries rather than from the normal program libraries used in your JCL. Parameters that allow you to specify the DD names and dataset names of these libraries are provided. For CICS DL/I support, DCRs and PCRs are loaded into memory by the CICS program loader. The DCR and PCR libraries, therefore, are made part of the DFHRPL concatenation.

To avoid conflict caused by DCRs or PCRs having the same name as existing CICS programs and PSBs, Bridge allows you to use alias names for DCRs and PCRs. If you give a DCR or PCR an alias, the alias takes precedence over the original name when Bridge loads it into memory.

First, Bridge allows you to use a standard or global DCR library where you can keep all production DCRs. Bridge provides only one global DCR library when installed. If you use a global DCR library, specify the dsname of this library in the DCRLIB parameter.

Bridge also permits the use of one or more local DCR libraries. These libraries might contain DCRs segregated by type (production and test), or they may be organized by application or based on your organization. If you decide to use local DCR libraries, you must specify the DD name that Bridge looks for in the DCRLIBDD parameter. You can use a global DCR library, one or more local DCR libraries, or a combination of both.

When Bridge initializes its IMS environment, it attempts to allocate a local DCR library defined in the job step's JCL. If that fails, it attempts to dynamically allocate the global DCR library. If that fails, the IMS Environment does not initialize.

Loading the PCR

A PCR corresponding to the PSB name used by an application program must exist. During initialization, Bridge retrieves the program PSB name and attempts to load a PCR by the same name. If the PCR is absent, a message is sent to the job log and the program is abnormally terminated.

As the PCB and the PCR have the same name, both cannot reside in the JOBLIB/STEPLIB/link list libraries. Bridge supports a special load module library called a PCR library. Always generate PCRs into PCR libraries.

Bridge locates the PCR library and loads the PCR in this sequence:

1. Bridge locates the local PCR library DD statement (the name of this DD statement is specified during Bridge installation). If this DD statement is not located or the PCR is not in the local PCR library, Bridge continues to the next step.

2. Bridge dynamically allocates the global PCR library (the name of this library is specified during Bridge installation). If this library cannot be allocated, or the PCR is not in the global PCR library, Bridge sends a message to the job log and terminates the program.

Loading the Bridge Routine

A corresponding Bridge Routine must exist for every DBD that requires bridging and is used by the application program. During initialization, Bridge retrieves the DBD name and attempts to load a Bridge Routine by the same name. If the Bridge Routine is absent, Bridge sends a message to the job log and the program is abnormally terminated.

The DBD and the Bridge Routine have the same name so both cannot reside in the JOBLIB/STEPLIB/link list libraries. Therefore, Bridge supports a special load module library called a DCR library. Always generate Bridge Routines into DCR libraries.

Bridge locates the DCR library and loads the Bridge Routine in this sequence:

1. Bridge locates the local DCR library DD statement (the name of this DD statement is specified during Bridge installation). If this DD statement is not located, or the Bridge Routine is not in the local DCR library, Bridge continues to the next step.

2. Bridge dynamically allocates the global DCR library (the name of this library is specified during Bridge installation). If this library cannot be allocated, or the Bridge Routine is not in the global DCR library, Bridge sends a message to the job log and terminates the program.

Managing PCR/Bridge Routine Copies in the MPP

If you create a new PCR or Bridge Routine while the MPP is active, an old copy may still reside in the MPP region. In order to enable immediate new PCR/Bridge Routine copies, stop and restart the MPP region.

Unsupported Functions

Direct processing on an Index DBD is not supported in this release of Bridge IMS Support.

Messages

All Bridge messages use WTOs to write to the job log. This means that the messages for the IMS DC are sent to the MPP job log.

Bridge uses an external message table you can customize during installation. (See the *ASG-Bridge Installation Guide*.)

User Abends

All Bridge user abends are accompanied by Bridge messages, and the abend codes exactly match the corresponding Bridge message numbers. If your IMS program abends with a user abend code, check the job log for a Bridge message with the same number.

Integrating Bridge into CICS DL/I Programs

Bridge provides forward and reverse bridging support in CICS programs that use EXEC DLI commands to access local DL/I databases (in the CICS address space) or DBCTL.

Note:

CICS DL/I support has the CICS Support feature as a co-requisite. Therefore, some functionality is common to both environments. For more information applicable to both environments, see "Integrating Bridge into CICS Programs" on page 145.

The process of defining and generating DBD Conversion Routines (DCRs) and PSB Conversion Routines (PCRs) is exactly the same as described in "Integrating Bridge into IMS Programs" on page 148 .

Aliases for DCRs and PCRs

In the CICS environment, DCRs and PCRs are defined and treated as CICS programs. As such, conflicts between the name of a CICS program and the name of a DCR or a PCR may occur. To avoid this conflict, Bridge CICS DL/I support provides for giving alias names to DCRs and PCRs. Use the alias name whenever the name of a DCR or a PCR is, or may already be, defined in your CICS system(s) as an existing program.

Specify the DCR alias name in the Bridge Definition on the Bridge Definition - Attributes screen. (See "Bridge Definition Attributes" on page 42.)

You assign the PCR alias name during the link-edit of the PCR into the PCR library. The Bridge CICS DL/I interface programs (VIAGFDLI and VIAGRDLI) receive a PCR name as a parameter. If you assign the PCR an alias name, this parameter must contain the PCR alias name.

CICS DL/I Program Integration

A CICS DL/I program triggers Bridge by calling one of the CICS DL/I Interface modules as shown in this table.

Module Name	Bridging Requirements
VIAGFDLI	A converted program that accesses an unconverted database.
VIAGRDLI	A unconverted program that accesses a converted database.

The call to Bridge requires the name of the PCR that is used in the bridging process as a parameter. If you assign the PCR an alias, this parameter must contain the PCR alias name.

This is the syntax for a COBOL program:

```
          .
          .
77  PCRNAME  PIC X(8)  VALUE 'ACCTPCR'.
          .
          .
          CALL 'VIAGFDLI' USING PCRNAME.
          .
          .
```

This is the syntax for a PL/I program:

```
          .
DCL  VIAGFDLI ENTRY OPTIONS(ASSEM, INTER, RETCODE);
DCL  PCRNAME  CHAR(8) INIT('ACCTPCR');
             .
             .
CALL VIAGFDLI (PCRNAME);
          .
```

This is the syntax for an assembler program:

```
          .
PCRNAME  DC  CL8'ACCTPCR'
          .
          .
          CALL VIAGFDLI,(PCRNAME),VL
          .
          .
```

Multiple PSB Support

Bridge supports CICS application programs that use multiple PSBs. The program must issue a call to the appropriate Bridge CICS DL/I interface module before each PSB it schedules, and specify the name of the PCR or its alias. No EXEC DLI calls are allowed between the Bridge initialization call and the SCHEDULE command.

Multiple Programs in a CICS Transaction

A CICS program may link or XCTL to other CICS programs. All programs in a single transaction should access IMS with either no bridging, forward bridging, or reverse bridging. A transaction does not support mixing programs with differing bridging requirements for the same PSB.

Accessing IMS and VSAM files

A single program can access both IMS and VSAM files. The program should issue a call to both the Bridge CICS VSAM initialization routine and the Bridge CICS DL/I initialization routine. Bridging of VSAM data and IMS data are independent of one another and are initialized and processed separately.

CICS ABEND Codes

See "CICS ABEND Codes" on page 146 for more information.

When Automatic Initialization of the Feature Fails

See "When Automatic Initialization of the Feature Fails" on page 162.

Manually Initializing the Bridge CICS DL/I Feature

The Bridge CICS DL/I feature can be initialized manually. To initialize the feature, enter the transaction `B2KL,ON` at a CICS terminal. If the feature fails to initialize properly, an appropriate message is issued.

Deactivating the Bridge CICS DL/I Feature

The CICS DL/I feature automatically deactivates as the CICS system shuts down. However, you can manually deactivate the Bridge CICS DL/I feature if necessary. To deactivate the feature, enter the transaction `B2KL,OFF` at a CICS terminal.

Note:

Use manual deactivation only in extreme situations. After you deactivate Bridge CICS DL/I, programs terminate with a 'B2K0' transaction abend code any time a DL/I bridging request is issued.

User Exit Routine

See "Integrating Bridge into CICS Programs" on page 145.

Defining DCRs and PCRs to CICS

You must define all DCRs and PCRs to CICS in the PPT by updating the CSD, by using CEDA transactions, or through AUTOINSTALL.

All DCR libraries and PCR libraries must be in the DFHRPL concatenation.

Integrating Bridge into IDMS Programs

Bridge provides forward and reverse bridging support for COBOL batch and online programs accessing IDMS databases. Bridge currently support IDMS versions 12 and 14.

Definition of Bridges

Bridge Definitions are defined for each IDMS schema containing records with fields to convert. Bridge generates an executable Bridge Routine for each Bridge Definition and a Subschema Conversion Routine for each subschema specified. These routines convert records during IDMS calls. One Bridge Definition is created for each schema and subschema that Bridge handles.

Bridge uses COBOL definitions of data structures that map to IDMS records. To simplify the definition process and avoid errors, Bridge uses the Bridge Definition in conjunction with the original and the new schema versions that contain records having fields in a new format to construct a Data Conversion Routine (DCR).

In a similar manner, Bridge uses information extracted from the IDMS Data Dictionary and the related DCRs to generate subschema Conversion Routines (SCRs) for each subschema.

IDMS Support Overview

A converted IDMS program triggers Bridge from a single call you add to the program initialization routine. This routine activates the IDMS run-time component of Bridge. After activation, Bridge intercepts all IDMS calls, handles the IDMS commands related to converted records, and uses the appropriate Bridge Routines to handle these commands:

- Retrieve Commands - Bridge intercepts the IDMS call, allows the retrieval to execute, then converts the retrieved record from its old format to its new format and presents the converted record to the program.
- Update Commands - Bridge intercepts the IDMS call, converts the record from its new format back to its old format and allows the update command to execute.

Using Bridge, a converted IDMS program can access an unconverted IDMS database as if it were converted. After the database converts, you can disable the Bridge Routine, allowing the application program to continue running without recompilation or further modification.

You initialize Bridge at the program level so that you can simultaneously access the same IDMS records from either converted or unconverted programs. Each program receives and processes the IDMS records in the format applicable to the program.

The Conversion Process - IDMS

To convert programs and databases, you perform several steps at the IDMS and program levels. Incorporating Bridge into a conversion project influences several steps in the process.

To perform a normal conversion and implement Bridge

1. Identify all fields in the database requiring conversion. This step is a direct outcome of the impact analysis.

2. Define a new version of the schema and subschemas to reflect the new format of the fields being converted. Bridge uses the new versions of the schema and subschemas to generate DCRs and SCRs. Your converted programs use these as well, but the IDMS system does not use them for accessing the physical databases.

3. Define a Bridge Definition for each schema, and generate the corresponding Bridge Routines (DCRs). This is done by using the Bridge Interactive Facility. See "Building a Bridge Definition" on page 41 and "Generating Executable Bridge Routines" on page 95.

4. For each subschema, generate the Subschema Conversion Routine (SCR). This is done by using the Bridge Interactive Facility. See "Generating IDMS Subschema Conversion Routines (SCRs)" on page 102.

5. Convert the application programs to refer to the new versions of the subschema and add a CALL to the program's initialization logic to trigger Bridge. Unit test the program and put it back into production.

6. After all programs are converted, you can convert the database and disable Bridge.

Describing the New Database Structure to IDMS

The situation which prevails when you prepare and use Bridge for the IDMS environment is different between the physical and logical database structure. This requires you to define all the relevant dictionary entities (schema, subschema, record, and element) once for the physical layout and once for the logical layout. Bridge uses the logical definition to compile bridged programs, and the physical definition to physically access IDMS.

That is, when you work with normal bridging, old (unconverted) programs still need the old database description. The new (converted) programs need the new description copied into the programs, but use the old subschema to access the database, since the actual database structure is still the old one.

Bridge IDMS support, as related to the IDMS dictionary, assumes that the old and new database definitions are simply different versions of the same schema and subschemas in the same dictionary.

The operation of the Bridge Interactive facility is based upon these assumptions:

- When you define a new Bridge Definition, the name you give the Bridge Definition should be the name of the schema.
- Specify old and new version numbers to Bridge to enable it to extract both the old and new subschema and record definitions from the data dictionary.
- Bridge uses the new version number to extract the subschema list, record name list and the record descriptions themselves from the IDMS data dictionary.
- When a DCR is generated, Bridge builds and submits JCL that refers to both versions of the schema and the records defined therein.
- Similarly, when generating an SCR, Bridge builds and submits JCL that refers to the schema name, subschema name, and the old and new version numbers.

Based on these assumptions, ASG recommends you define your new database as a new version of the existing schema. The new version number may be either higher or lower than the old version number depending on your site's standards.

Defining Database Conversion Routines

Bridge converts IDMS records from one format to another based upon the fields being converted and their old and new formats. To accomplish this conversion, Bridge needs information on all of the fields in the database being converted and the records where they are contained. These records and their respective fields are defined to Bridge in a Bridge Definition.

Bridge Definition

The Bridge Definition for an IDMS schema is similar to those of Sequential/VSAM files, with a few significant differences. For each schema Bridge handles, you create a corresponding Bridge Definition that defines all records containing fields being converted. To simplify this process, Bridge retrieves a list of record names from the IDMS Data Dictionary (IDD), and permits you to indicate what records in the list the DCR will support.

Bridge then retrieves COBOL language record structure definitions from the IDD for each selected record. Bridge automatically associates the IDMS Record Name to its respective COBOL data structure. Then you identify the Fields you want converted (Bridge Fields) and associate them to Bridge Rules which specify how the field will convert. Bridge assumes that all remaining fields remain the same between the old and new versions of each record.

After the Bridge Definition is complete, Bridge generates an executable Bridge Routine [a Database Conversion Routine (DCR)] containing all of the information and logic needed to convert each record defined. This generate process retrieves and compares the old and new versions of all records in the schema and ensures that all changes or differences are addressed in the Bridge Definition.

Generating the Bridge Routines: DCRs and SCRs

After you define the Bridge Definition for the schema, you can generate the Bridge Routines. Bridge scans both the original and new versions of the schema in conjunction with the Bridge Definition, and creates a load module called a DCR that contains routines and information required to efficiently identify and convert data.

These routines and information comprise the generated DCR:

- I/O conversion routines to convert entire records
- Field conversion routines to convert specific fields
- Information on all records in the schema requiring conversion
- Information on all fields in each record requiring conversion

Any time the schema is modified or bridging requirements change for one or more records in the schema, you must regenerate a DCR.

After you generate the DCR, you can generate an SCR for one or more of the schema's subschemas.

The generated SCR is composed of information on all records in the subschema. Any time the subschema or schema is changed, you must regenerate a SCR.

IDMS Program Integration

A converted IDMS program triggers Bridge by calling one of the Bridge initialization routines. After Bridge initializes, it intercepts and interrogates all IDMS calls. Those IDMS calls that result in reading or writing IDMS records are processed by Bridge, bridging the record as specified by the applicable SCR and DCR.

During Bridge initialization, these preliminary preparations of the bridging environment take place:

- The SCRs designated by your program are loaded as well as the appropriate DCRs.
- A data structure is built in memory according to the SCR/DCR to reflect the records and fields requiring bridging.

After Bridge initializes, it intercepts and analyzes each IDMS call to determine its bridging relevancy (e.g. OBTAIN, STORE requests).

- The DCR routines are invoked to convert the dates in the fields described by the SCR modules.
- All this is done according to the desired bridging type, namely the forward bridging or reverse bridging for batch and online programs.

Invoking Bridge in an IDMS Program

To trigger Bridge you must make two changes to a given IDMS program:

1 Change the PROTOCOL statement from the existing protocol to the Bridge IDMS protocol as shown:

Current Protocol	Bridge Protocol	Remarks
BATCH	VIAGIDB	Translates a DML verb into a VIAGIDMS call instead of an IDMS call
BATCH AUTOSTATUS	VIAGIDB-AUTO	Translates a DML verb into a VIAGIDMS call instead of an IDMS call and does automatic validation of the IDMS ERROR STATUS field
IDMS-DC-NONAUTO	VIAGIDC	Translates a DML verb into a VIAGIDC call instead of an IDMSCOBI call
IDMS-DC	VIAIDC-AUTO	Translates a DML verb into a VIAGIDC call instead of an IDMSCOBI call and does automatic validation of the IDMS ERROR STATUS field

Note:

The DEBUG option on the PROTOCOL clause is also supported as are all other options.

2 Add a call to the Bridge initialization routine as shown:

Call	Batch	Online
Forward Bridging	VIAGFIDM	VIAGFIDC
Reverse Bridging	VIAGRIDM	VIAGRIDC

The syntax for this call is:

```
CALL 'VIAGxxxx' USING SUBSCHEMA-CTRL, SCRNAME
```

where:

SUBSCHEMA-CTRL is the address of the SS-CTRL (subschema control record) and *SCRNAME* is the name of the Bridge SCR associated with the subschema.

The call to the Bridge initialization routine may contain one or more pairs of SUBSCHEMA-CTRL, SCRNAME, for example:

```
CALL 'VIAGRIDM' USING SUBSCHEMA-CTRL1, SCRNAME1,
SUBSCHEMA-CTRL2, SCRNAME2.
```

- The protocol must be changed in each converted IDMS program that uses Bridge.
- Because of the change in the IDMS call caused by using the Bridge protocol, Bridge intercepts all IDMS calls. However, the calls are passed through to IDMS unaffected by Bridge in any way unless and until the call to the Bridge initialization module takes effect in a Bridge Routine.
- The call to the Bridge initialization routine should be added in every program with an IDMS BIND RUN-UNIT statement. The call must be included in the program such that it executes before the first IDMS call (BIND RUN-UNIT), and before any other DML call, including IDMS-DC DML calls such as MAP IN, etc.
- A routine with an IDMS BIND RUN-UNIT statement must include the call to the Bridge IDMS interface.
- A routine with an extended run-unit (uses a SS-CTRL passed from the main program) does not need to call the Bridge initialization routine, but the PROTOCOL statement must be changed to VIAGBTCH, the same protocol as in the originating program.

Combining IDMS and Sequential/VSAM Bridging

The Bridge IDMS interface coexists with other Bridge interfaces (such as the Bridge Sequential/VSAM interface), yet operates independently. If you want to activate multiple Bridge interfaces in your program, include calls to each respective Bridge initialization routine. You can activate any Bridge interface first, followed by the other. Activating one does not affect the other, regardless of whether it is active.

Restrictions

- Bridge IDMS supports the non-SQL database structure form only.
- Logical Record Facility (LRF) is not supported.
- Keys (CALC keys and sort keys) are supported only through direct access. For example, when reading a set sequentially, no guarantee is given as to the order of the records retrieved.
- In a schema, some records may be converted and some not, but if a record is converted, all bridged programs access the converted record.

Run-time Issues

DCR and SCR Libraries

The IDMS Support Option of Bridge constructs and uses two types of executable Bridge Routines: Database (schema) Conversion Routines (DCRs) and Subschema Conversion Routines (SCRs). A DCR contains bridging routines and information about the schema that it converts. An SCR contains bridging routines and information about the subschema that it converts.

DCRs and PCRs are named the same as the schema that they convert. As DCRs or SCRs may have the same name as one or more of your programs, you must treat DCRs and SCRs differently than programs to prevent conflict when trying to load one or the other. Bridge uses special libraries for DCRs and SCRs called DCR Libraries and SCR Libraries, respectively.

Specifying the IDMS DCR and SCR Libraries for Batch Programs

For IDMS support, Bridge loads DCRs and SCRs from their respective libraries rather than from the normal program libraries used in your JCL. Parameters that allow you to specify the DD names and dataset names of these libraries are provided and specified when you install Bridge.

First, Bridge allows you to use standard or global DCR and SCR libraries. At installation, Bridge provides only one global DCR library and one global SCR library. If you use either or both global libraries, you must specify the dataset name(s) of each during installation of Bridge (see the DCRLIB and SCRLIB parameters in the *ASG-Bridge Installation Guide*).

Bridge also allows you to use one or more local DCR and SCR libraries. These libraries might contain DCRs or SCRs segregated by type (production vs. test), or they may be organized by application or based upon your organization. If you decide to use local DCR and SCR libraries, you must specify the DD names that Bridge looks for when allocating the local DCR or SCR library.

You can use a global DCR library, one or more local DCR libraries, or a combination of both. The same is true of SCR libraries.

Loading the SCR for Batch Programs

Recalling that your program tells Bridge what SCRs to use for bridging, the SCR(s) you specify must be in the local and/or global SCR libraries available to your program at execution time. Otherwise a Bridge run-time error is signaled and your program terminates.

Bridge locates the SCR libraries and loads the SCR in this sequence:

1. Bridge locates the local SCR library DD statement (the name of which was specified when Bridge was installed). If this DD statement is not located in your program's JCL, or the SCR is not contained in the local SCR library, Bridge continues to the next step.

2. Bridge dynamically allocates the global SCR library (the name of which was specified when Bridge was installed). If this library is not allocated, or the SCR is not contained in the global SCR library, Bridge issues an error message to the job log and terminates the program.

Loading the DCR for Batch Programs

A DCR must exist for each schema used by the program. Bridge retrieves the DCR name from the active SCR, and attempts to load the DCR into memory. If the DCR cannot be found, a message is written to the job log and the program is terminated.

DCRs are located and loaded into memory in this sequence:

1. Bridge locates the local DCR library DD statement (the name of which was specified when Bridge was installed). If this DD statement is not located in your program's JCL, or the DCR is not contained in the local DCR library, Bridge continues to the next step.

2. Bridge dynamically allocates the global DCR library (the name of which was specified when Bridge was installed). If this library is not allocated, or the DCR is not contained in the global DCR library, Bridge issues an error message to the job log and terminates the program.

Loading DCRs and SCRs for online programs

For IDMS-DC support, Bridge loads DCRs and SCRs from their respective libraries in the IDMS program pool with an IDMS LOAD command (instead of an MVS LOAD command).

The SCR, by default, carries the same name as the subschema. Because the default search order in the IDMS environment is load area followed by load libraries, this IDMS LOAD command is parameterized to load the SCR from the load libraries (concatenated in the CDMSLIB DD statement of the Central version) instead of the IDMS load areas. This keeps Bridge from mistaking a subschema for a SCR.

However, some IDMS customers punched their subschema load modules from the load area and linked them in a load library. In this case, ASG recommends that you change the SCR name to a different name and use the new name in the Bridge initialization statement in the COBOL program. The SCR generation process allows you to specify an alternative name for the SCR.

You must define all DCRs and SCRs in the IDMS system generation as program entities (with the language ASSEMBLER) to avoid IDMS message DC021002.

Unsupported Functions

These functions or features are not supported in this release of Bridge IDMS support.

- SQL
- Logical Record Facility (LRF)

Messages

All Bridge messages are written to the:

- Job log by using WTOs for batch programs
- IDMS DDLDCLOG area for online programs

User Abends

All Bridge User Abends are accompanied by Bridge messages, and the abend codes exactly match the corresponding Bridge message numbers. Therefore, if you IDMS program abends with a user abend code, be sure to check the job log or the DDLDCLOG area for a Bridge message with the same number.

Reverse Bridging for All Environments

Reverse Bridging is the process of Bridge running backwards, and is a standard feature of Bridge. Reverse Bridging is used to process files converted to a new format (e.g., date fields expanded) with a program that has not been converted to process them.

You integrate Reverse Bridging into a program in the same manner as Forward Bridging, with one exception. The name of the Bridge Initialization Module called by a program is different. This table lists the Reverse Bridging Initialization Routines:

Routine Name	Where Used
VIAGRASM	Assembler Language Programs
VIAGRCOB	COBOL Programs
VIAGRPLI	PL/I Programs
VIAGRCIC	CICS Programs, regardless of language
VIAGRIMS	IMS Programs, regardless of language
VIAGRDLI	DL/I Programs, regardless of language
VIAGRIDM	COBOL IDMS batch programs
VIAGRIDC	COBOL IDMS online programs

12 Record-Level API

This chapter describes the Record-Level Application Programming Interface (RLAPI) and contains these sections:

The RLAPI feature lets you use Bridge in situations where the standard run-time facility does not support the application or environment that needs bridging. Situations that may require the RLAPI include:

- Your program uses an access method that is not currently supported by Bridge. Examples of this are BSAM and EXCP, where the user program is responsible for record blocking and deblocking.
- Your program runs in an environment that is not supported by Bridge. Examples of this are commercial and custom database management systems not supported by Bridge.
- You do not want to use the standard Bridge run-time engine.

The RLAPI interfaces directly with Bridge Routines, completely bypassing the Bridge run-time facility. Thus, the open and read/write intercepts provided by Bridge, which normally make bridging automatic and transparent to the user program, are not used. However, all of the facilities of dynamic bridging performed by Bridge Routines, the automatic disabling of a Bridge Routine, and the run-time trace facility are all still available.

A program that uses the RLAPI identifies when records are bridged, what Bridge Routine is used, and whether forward or reverse bridging is activated. You must insert calls to the API into your program at the appropriate points to bridge records immediately after they are read and immediately before they are written.

The RLAPI presents a return code indicating the success or failure of an RLAPI call. A user program should examine the RLAPI return code and respond in a manner applicable to the application and situation. To assist in diagnosing a problem, a Reason Code returns indicating the cause of the problem.

Using the RLAPI

To prepare your program to use the RLAPI

1. Identify each of the files read and/or written by your program that require bridging. Typically, they are files that require conversion because of changes in certain fields (usually currency or date fields).

2. Identify the most appropriate points in your program to insert the Bridge RLAPI calls. In the case of records being read from files, this is usually after the record is read and before the record is processed by the program.

   If the program is deblocking records, insert the RLAPI call after the deblocking occurs, when the program identifies the logical record to process. In the case of written records, insert the RLAPI call after all record processing completes and before the record actually writes.

   If the program is blocking records, insert the RLAPI call before blocking occurs, while the program can still identify the logical record.

3 Insert an RLAPI call at each point identified in the previous step. For COBOL, call the VIAGFAPI module for forward bridging, or call the VIAGRAPI module for reverse bridging. The call passes parameters to the RLAPI specifying, among other things, the record to bridge and where the RLAPI places the bridged record. See "RLAPI Parameters" on page 175 for further information.

Call	Forward Bridging	Reverse Bridging
COBOL/REXX	VIAGFAPI	VIAGRAPI
PL/I	VIAGFAP	VIAGRAP
CICS COBOL (see note)	VIAGCAPI	VIAGCAPI

Note:

The VIAGCAPI module is the same for forward and reverse bridging. You must specify either F(orward) or R(everse) on the Direction parameter.

4 Immediately after the RLAPI call, insert coding to evaluate the return code from the RLAPI. This return code indicates the success or failure of the call, and the program responds to the return code in a manner appropriate to your application. See "RLAPI Error Analysis" on page 176 for more information.

RLAPI Parameters

Parameters pass to the RLAPI in the form of a standard MVS parameter list of fullword addresses, each containing a pointer to the actual parameter. The last item in the list is designated by the high-order bit setting of one. At RLAPI entry, register one must point to the beginning of the parameter list. Most language compilers generate a parameter list meeting these requirements.

This is the list of the parameters:

Parameter	Description
The Input record	This is the record that requires bridging
Output area	This is the area where the bridged record is placed when the bridging operation is completed
Input record length	This is a binary halfword containing the length of the input record

Parameter	Description
Output area length	This is a binary halfword containing the length of the output area.
Bridge Routine name	This is an eight-byte character field that contains the name of the Bridge Routine that is used. If the name is less than eight characters long, it must be left-justified and right-filled with blanks.
Feedback area	This is a four-byte area where the RLAPI returns error codes used for error analysis by the program. See "RLAPI Feedback Area" on page 177 for further information.
Anchor Word for Bridge RLAPI use	This must be a fullword, initialized to binary zeros during program initialization, and untouched by the program thereafter. The RLAPI uses this word as an anchor point for the storage that it uses during the execution of your program.

Additional parameters for CICS COBOL program call the same module for forward and reverse bridging. In VIAGCAPI, you must specify F(orward), or R(everse) on the Direction parameter as shown in this syntax:

```
77  DIRECTION      PIC X     VALUE 'F'.
```

Note:

See the example of the RLAPI call in "CICS - COBOL" on page 181.

RLAPI Error Analysis

The RLAPI error analysis places a return code into Register 15 when the RLAPI returns to your program. Language compilers may make the contents of this register available to the programmer in some other fashion, for example:

- COBOL places the return code into the COBOL special register RETURN-CODE that the programmer can examine and reset.
- PL/I places the return code in the function PL/I RETC.

The return code indicates whether the bridging operation succeeded or failed. The program examines the return code and responds appropriately.

The RLAPI return code is also placed in the Return Code field of the RLAPI Feedback Area. The program may test the return code there as an alternative to testing the return code in register 15.

To facilitate simple error analysis in a program, the error analysis keeps the number of return codes to a minimum. Each code has a distinct meaning as shown in the RLAPI Return Codes chart. However, a number of things can trigger a specific return code. The person responsible for error diagnosis needs to know the value of the Reason Code to properly diagnose the exact cause of an error. The cause of the error displays in the Reason Code field in the RLAPI Feedback Area to aid in diagnosis. The program can examine the Reason Code and act on it accordingly, or you may decide to simply display or print an error message containing the Reason Code value for further analysis. See "RLAPI Reason Codes" on page 179 for detailed descriptions.

RLAPI Feedback Area

The RLAPI Feedback Area is a four-byte area containing a pair of two-byte binary fields; the RLAPI Return Code and Reason Code. The is the format of the Feedback Area:

+0 - Return Code	+2 - Reason Code

The Return Code field contains the same value that is returned in register 15 as a return code.

The Reason Code field contains a numeric value indicating the exact cause of the error or the unusual condition detected. The Reason Code value is unique to the Return Code value. That is, Reason Code 1 accompanying Return Code 4 has a meaning different from Reason Code 1 accompanying Return Code 12. See the RLAPI Return Codes and "RLAPI Reason Codes" on page 179 for further information.

RLAPI Return Codes

The RLAPI returns a code indicating the success or failure of the RLAPI call. These are the RLAPI return codes, reason code values, and their respective meanings:

Code	Meaning/Recommended Action
0	Successful Operation No errors occurred, no action required Reason Code contains zero.
4	Bridging operation successful, but unusual condition detected. Reason Code indicates condition
8	Bridging operation not attempted Incomplete or invalid parameter list detected by RLAPI (this is a programming error) Parameter list is missing, or contains too few or too many parameters No information is placed in Feedback Area.
12	Bridging operation not attempted Parameter error detected by RLAPI Reason Code indicates exact cause of error
16	Bridging operation failed Error detected in Bridge Routine Reason Code indicates exact cause of error.

RLAPI Reason Codes

These are the RLAPI reason codes:

Return Code	Reason Code	Meaning/Description
4	1	Bridge Routine indicated File is converted Record did not need bridging. Input record moved to output area as-is
12	1	Input Record Address is zero
	2	Output Area Address is zero
	3	Input Record Length is zero
	4	Output Record Length is zero
	5	Bridge Routine name missing or invalid, or Bridge Routine module could not be located or loaded successfully.
	6	Bridge anchor word is not the last parameter of the parameter list
	7	A GETMAIN of a work area failed due to insufficient storage
16	4	Input Record too short. Bridge Routine expecting longer record than program is providing VIAG-0620 message printed in Message Log
	10	Bridge Routine encountered unrecognized Record Type value Error accompanied by VIAG-0616 message in Message Log
	16	Output Area too short Routine requires output area longer than program is providing Error accompanied by VIAG-0619 message in Message Log

RLAPI Examples

COBOL

This is an example of an RLAPI call in a COBOL program:

```
.
.
WORKING-STORAGE SECTION.
.
.
01   BRIDGE-INPUT-AREA    PIC X(240).
01   BRIDGE-OUTPUT-AREA.
     COPY MASTER.
.
77 BRIDGE-INPUT-LENGTH          PIC S9(4)  COMP VALUE 240.
77 BRIDGE-OUTPUT-LENGTH   PIC S9(4)  COMP.
77   BRIDGE-ROUTINE-NAME  PIC X(8)   VALUE 'BRGMSTR'.
01 BRIDGE-FEEDBACK-AREA.
     05  BRIDGE-RETURN-CODE    PIC S9(4)  COMP.
     05  BRIDGE-REASON-CODE    PIC S9(4)  COMP.
77   BRIDGE-ANCHOR-WORD   PIC S9(8)  COMP VALUE 0.
.
.
PROCEDURE DIVISION.
.
.
READ-MASTER SECTION.
*
*    Read the Master File into the Bridge input area.
*
     READ MASTER-FILE INTO BRIDGE-INPUT-AREA
     AT END
          MOVE HIGH_VALUES TO MASTER-EOF-SWITCH
          GO TO READ-MASTER-EXIT.
*
*    Tell Bridge how long the Bridge Output Area is. This is necessary
*    because Bridge returns the length of the resulting output record in
*    this same area. Some applications need that information. We do not.
*
     MOVE 256 TO BRIDGE-OUTPUT-LENGTH.
*
*    Now call Bridge and have it expand the record.
*
     CALL 'VIAGFAPI' USING
          BRIDGE-INPUT-AREA,
          BRIDGE-OUTPUT-AREA,
          BRIDGE-INPUT-LENGTH,
          BRIDGE-OUTPUT-LENGTH,
          BRIDGE-ROUTINE-NAME,
BRIDGE-FEEDBACK-AREA.
BRIDGE-ANCHOR-WORD.
*
*    The return code is in the COBOL Special Register "RETURN-CODE".
*    Check it for errors.  I could also have used "BRIDGE-RETURN-CODE".
*  If an error occurred, display the return code and reason code and
*  terminate the program.
*
```

```
           IF RETURN-CODE > 0
           DISPLAY '** WARNING: Bridge Failed to bridge a MASTER record'
           DISPLAY '*** Return Code from Bridge is ' RETURN-CODE
           DISPLAY '*** Reason Code from Bridge is ' BRIDGE-REASON-CODE
           STOP RUN.

      READ-MASTER-EXIT.
           EXIT.
```

CICS - COBOL

This is an example of an RLAPI call in a CICS/COBOL program:

```
       IDENTIFICATION DIVISION.
        PROGRAM-ID. VIAGCRIV.
        ENVIRONMENT DIVISION.
        DATA DIVISION.
        WORKING-STORAGE SECTION.

        77  DIRECTION      PIC X     VALUE 'F'.
        77  IN-LENGTH      PIC S9(4) COMP VALUE 80.
        77  OUT-LENGTH     PIC S9(4) COMP VALUE 80.
        77  BRIDGE-ROUTINE PIC X(8)  VALUE 'VIAGCDCF'.
        01  FEEDBACK.
            05  FEED-1     PIC S9(4) COMP VALUE 0.
            05  FEED-2     PIC S9(4) COMP VALUE 0.
        77  ANCHOR         PIC S9(8) COMP VALUE 0.

        01  NEW-LINE.
            05  FILLER     PIC X(1)  VALUE ' '.
            05  FILLER     PIC X(1)  VALUE ' '.
        01  HDDR-1.
            05  FILLER     PIC X(80) VALUE 'RECORD LEVEL API IVP'.
        01  PRESMSG.
            05  FILLER     PIC X(1)  VALUE ' '.
            05  FILLER     PIC X(1)  VALUE ' '.
            05  FILLER     PIC X(79) VALUE 'INPUT RECORD  '.
            05  DISP-IN    PIC X(79) VALUE SPACES.
            05  FILLER     PIC X(79) VALUE 'OUTPUT RECORD '.
            05  DISP-OUT   PIC X(79) VALUE SPACES.
            05  FILLER     PIC X(79) VALUE 'PROCESSING COMPLETED'.
            05  FILLER     PIC X(79) VALUE SPACES.
        01  IN-RECORD.
            05  FILLER     PIC X(5)  VALUE 'AAAAA'.
            05  DATE-IN    PIC X(6)  VALUE '960323'.
            05  FILLER     PIC X(69) VALUE 'BBBBBB'.
        01  OUT-RECORD.
            05  FILLER     PIC X(5)  VALUE '     '.
            05  DATE-OUT   PIC X(8)  VALUE '      '.
            05  FILLER     PIC X(67) VALUE SPACES.

        PROCEDURE DIVISION USING DFHEIBLK DFHCOMMAREA.

            MOVE SPACES TO OUT-RECORD.
          EXEC CICS SEND TEXT FROM(HDDR-1) LENGTH(20) ERASE END-EXEC.

          MOVE SPACES TO OUT-RECORD.

          CALL 'VIAGCAPI' USING
               DFHEIBLK,
```

```
                DFHCOMMAREA,
                DIRECTION,
                IN-RECORD,
                OUT-RECORD,
                IN-LENGTH,
                OUT-LENGTH,
                BRIDGE-ROUTINE,
                FEEDBACK,
                ANCHOR.

            MOVE DATE-IN      TO DISP-IN.
            MOVE DATE-OUT     TO DISP-OUT.

            EXEC CICS SEND TEXT FROM(PRESMSG) LENGTH(410) END-EXEC.
        FIN.
            EXEC CICS RETURN END-EXEC.
            GOBACK.
```

PL/I

This is an example of an RLAPI call in a PL/I program:

```
/*****************************************************************/
/*  SAMPLE PL/I PROGRAM FOR BRIDGE RECORD LEVEL APPLICATION      */
/*  PROGRAM INTERFACE(RLAPI).  SAMPLE OF EURO CALL.              */
/*****************************************************************/

SAMPPLI : PROC ;                                              00180
     DCL VIAGFAP             EXT ENTRY OPTIONS (ASM,RETCODE,INTER);
     DCL 1 CURRENCY_RECORD,
           5 CURRENCY_ID               CHAR (3) INIT('EUR'),
           5 NATIONAL_CURRENCY         PIC '(7)9V99' INIT(0),
           5 EURO_CURRENCY             PIC '(7)9V99' INIT(0),
           5 ROUND_LOSS                PIC '(4)9V99' INIT(0);

     DCL REDEF_RECORD   BASED(ADDR(CURRENCY_RECORD)) CHAR(27);

     DCL BRIDGE_INPUT_LEN              FIXED BIN(15) INIT(27);
     DCL BRIDGE_OUTPUT_LEN             FIXED BIN(15) INIT(27);
     DCL 1 BRIDGE_FEEDBACK_AREA,
           5 BRIDGE_RC                 FIXED BIN(15) INIT(5),
           5 BRIDGE_RESC               FIXED BIN(15);
     DCL BRIDGE_ANCHOR_WORD            FIXED BIN(31) INIT(00);

     DCL DUPLICATE_AREA                CHAR(27) INIT (' ');

     DCL NAME_BRIDGE                   CHAR(8) INIT ('TESTBRD ');

     /*    EXECUTABLE INSTRUCTIONS                                */
     DUPLICATE_AREA = REDEF_RECORD;
     PUT SKIP LIST('BEFORE BRIDGE ']]DUPLICATE_AREA);

     CALL VIAGFAP(DUPLICATE_AREA,        /* PARM 1 */
                  CURRENCY_RECORD,       /* PARM 2 */
                  BRIDGE_INPUT_LEN,      /* PARM 3 */
                  BRIDGE_OUTPUT_LEN,     /* PARM 4 */
                  NAME_BRIDGE,           /* PARM 5 */
                  BRIDGE_FEEDBACK_AREA,  /* PARM 6 */
                  BRIDGE_ANCHOR_WORD);   /* PARM 7 */
```

```
/*  THIS CALL CONVERTS ZONED DECIMAL FIELD CALLED           */
/*  NATIONAL_CURRENCY TO A ZONED DECIMAL FIELD CALLED       */
/*  EURO_CURRENCY AND STORES DIFFERENCE IN ROUND_LOSS       */
/*  USING CURRENCY_ID TO DETERMINE THE NATIONAL CURRENCY.   */
/*  PARM1 = NAME OF RECORD TO BE CONVERTED                  */
/*  PARM2 = NAME OF NEW RECORD                              */
/*  PARM3 = LENGTH OF THE RECORD TO BE CONVERTED            */
/*  PARM4 = LENGTH OF THE RECEIVING RECORD                  */
/*  PARM5 = BRIDGE RULE TO BE USED FOR CONVERSION           */
/*  PARM6 = RETURN CODE INFO FROM CONVERSION                */
/*  PARM7 = FLAPI ANCHOR POINT (NEVER MODIFIED BY USER)     */

   PUT SKIP LIST('BRIDGE MESSAGE ']]BRIDGE_FEEDBACK_AREA);
   PUT SKIP LIST(CURRRENCY_RECORD);

END SAMPPLI;
```

REXX

This is an example of an RLAPI call in a REXX program:

```
/*REXX*/
LINLINE=D2C(80)              /* INPUT RECORD LENGTH         */
/* OUTLINE ( MUST BE LONG ENOUGH TO HOLD CONVERTED RECORD ) */
OUTLINE=RIGHT(' ',86)
LOUTLINE=D2C(86)             /* OUTPUT RECORD LENGTH        */
ANCHOR=X2C('00000000')       /* BRIDGE ANCHOR WORD          */
BROUTINE="VIAGCDCB"          /* BRIDGE CONVERSION ROUTINE   */
FDBKAREA=X2C('00000000')     /* BRIDGE FEEDBACK AREA        */
"ALLOC FI(IN) DA('VIASOFT..&CENTER..CNTL(VIAGDMG)') SHR REUSE"

"EXECIO * DISKR IN (FINIS STEM IN."

 DO I = 1 TO IN.0
   INLINE = IN.I
   ADDRESS LINKPGM "VIAGFAPI INLINE OUTLINE LINLINE ",
                   "LOUTLINE BROUTINE FDBKAREA ANCHOR"
/*  PARM1 = NAME OF BRIDGE DRIVER ( VIAGFAPI OR VIAGRAPI )  */
/*  PARM2 = NAME OF RECORD TO BE CONVERTED                  */
/*  PARM3 = NAME OF NEW RECORD                              */
/*  PARM4 = LENGTH OF THE RECORD TO BE CONVERTED             */
/*  PARM5 = LENGTH OF THE RECEIVING RECORD                  */
/*  PARM6 = BRIDGE RULE TO BE USED FOR CONVERSION           */
/*  PARM7 = RETURN CODE INFO FROM CONVERSION                */
/*  PARM8 = RLAPI ANCHOR POINT (NEVER MODIFIED BY USER)     */

   IN.I = OUTLINE

   SAY "INLINE  =" INLINE
   SAY "OUTLINE =" OUTLINE
   SAY "FDBK    =" C2X(FDBKAREA)
 END

RETURN
```

Debugging RLAPI problems

You may encounter problems when testing your program changes and the RLAPI. Use these tips when debugging:

- The RLAPI Return Codes and Reason Codes are designed to help you. Use the charts provided on page 178 to determine the cause of the problem.
- Use the Bridge Trace facility with the RLAPI if you think the RLAPI is not working correctly (see "Run-Time Debugging" on page 143). Be sure to remove the VIAGTRAC DD statement from the JCL when you no longer need it.
- Problems relating to the parameter list are almost always encountered during the first few attempts to test the RLAPI. Examine the call and the parameter list carefully to ensure it is coded it correctly.
- Verify that an input record is bridged before the program begins to process it. Otherwise, the program may attempt to process a record that is not yet bridged. Unpredictable results can occur.
- Verify that all processing of an output record is completed before bridging an output record. Otherwise, changes may be made to the record after bridging that do not display in the bridged record.

Use VIAGTEST to test the Bridge Routine outside of the program. (See "Testing Your Bridge Routine" on page 131 for more information about VIAGTEST.) VIAGTEST helps you to determine if the problem is in the Bridge Routine or the program.

Field Level Support

In Bridge 6.0, the RLAPI supports field level bridging. It is easier to implement RLAPI on a field basis than it is to use the FLAPI provided with previous Bridge releases. Components of FLAPI remain in the product libraries and existing implementations will continue to function.

Appendix A

Bridge Field and Record Definition Association Errors

These are the Bridge field and record definition association errors:

	Error Description	Suggested Solution
1	Bridge Fields have Record Definition associations that overlap or are subsets. This error occurs when the association to one of the Bridge Fields is incomplete.	Add associations to the Bridge Field that is incomplete. Reset the Bridge Field; however, this action would prevent the field from being converted. Delete associations so that the two fields have equivalent associations.
2	Bridge Fields share a lowest level Record Definition association, but in further associations, one field is missing an association that the other has.	Associate the Bridge Field with the missing Record Definition.
3	The Bridge Fields listed not associated to a Record Definition of the lowest level currently in use. If no Bridge Fields are listed, the Record Definition is incorrectly defined by not including the lowest Record Definition level.	Associate the Bridge Field with the Record Definition of the lowest level currently in use.

	Error Description	Suggested Solution
4	Multiple record layouts exist and some Bridge Fields are defined, but no Record Definitions are defined. Consequently, no associations exist.	If the source contains multiple record formats (01-levels in COBOL), a Record Definition must be defined to Bridge for identification of the record formats to determine what fields to bridge. Define at least one Record Definition per record format, and associate the appropriate Bridge Fields with the Record Definition.
5	Bridge Fields from two different record formats are associated to the same Record Definition.	Bridge Field Associations cannot overlap. Each Record Definition identifying a record format can be associated only to Bridge Fields from the same record format. Delete the association to the Bridge Field from a different record format.

Appendix B
Program Messages and User Abends

Batch Messages

Bridge user abend codes are the same as the four-digit message number. These are the program messages and user abends:

Message	Description and Explanation	System Action	User Action
VIAG-0101-E	ERROR IN DATE CONVERSION ROUTINE PARAMETER User program called the Bridge initialization routine (VIAGFCOB.etc) with wrong number of parameters. Parameters should be in pairs of ddname, DCR.	Program abends.	Correct parameter list and rerun job.
VIAG-0102-E	ERROR IN DD NAME PARAMETER: <parameter> Invalid DD name was passed to Bridge initialization routine.	Program abends.	Correct problem and rerun job.
VIAG-0103-E	REQUESTED LOAD MODULE <name> DOES NOT EXIST DCR specified was not found or cannot be loaded.	Program abends.	Correct name or generate DCR.
VIAG-0104-E	BRIDGE PARAMETERS MISSING No parameters were specified in call to Bridge initialization routine.	Program abends.	Add parameters list to Bridge initialization routine and rerun job. Parameters list should contain ddnames and their DCR names for Bridge to handle.

Message	Description and Explanation	System Action	User Action
VIAG-0105-E	LOADING OF INIT USER EXIT <exitname> FAILED Init user exit name, as specified in VIAGBRGP module, was not found or cannot be loaded.	Program abends.	Determine why user exit cannot be found, or remove user exit name from VIAG BRGP module (Bridge installation parameter).
VIAG-0106-E	LOADING OF OPEN USER EXIT <exitname> FAILED Init user exit name, as specified in VIAGBRGP module, was not found or cannot be loaded.	Program abends.	Determine why user exit cannot be found, or remove user exit name from VIAG BRGP module (Bridge installation parameter).
VIAG-0107-E	INVALID FILE ,NULLFILE OR OS FILE. DD:<ddname> Specified DD name should point to VSAM file, but file is OS or nullfile according to user program.	Program abends.	Correct problem and rerun job.
VIAG-0108-E	SHOWCAT FAILED. DD:<ddname> , RC: <return code> Read catalog data failed.	Program abends.	Verify return code that displays in message. Correct problem and rerun job.
VIAG-0109-E	SHOWCAT READ DATA RECORD FAILED. DD:<ddname> Read catalog data failed	Program abends.	Correct problem and rerun job.
VIAG-0111-E	INVALID DBMS FOR OS/VSAM FILE. DCR:<bridgeroutinename> Invalid DBMS specified in DCR. DBMS should be a none for VSAM/OS file.	Program abends.	Correct DBMS parameter and generate DCR.
VIAG-0120-I	REVERSE BRIDGING IS ACTIVE Program is running with reverse bridging option for VSAM/OS files.	None.	None.

Message	Description and Explanation	System Action	User Action
VIAG-0201-E	LOADING OF BRIDGE GENERAL PARAMETERS FAILED Bridge general parameters module (VIAGBRGP) was not found or cannot be loaded.	Program abends.	Verify that Bridge load library is located in steplib on linklist and rerun job.
VIAG-0202-E	DD NAME PARM MISSING IN VIAGRTO CARD Syntax error in VIAGRTO cards. Card should start with DD name.	Program abends.	Correct statement in VIAGRTO file and rerun job.
VIAG-0203-E	STATUS PARM MISSING IN VIAGRTO CARD Status parameter (FILE-CONVERT) missing.	Program abends.	Correct statement in VIAGRTO file and rerun job. Syntax is DD name, FILE-CONVERTED = yes/no.
VIAG-0204-E	DD NAME IN VIAGRTO CARD IS INVALID Invalid DD name supplied in VIAGRTO file.	Program abends.	Correct problem and rerun job.
VIAG-0205-E	INVALID CONVERT REQUEST IN VIAGRTO CARD FOR DD <ddname> Invalid convert request supplied for FILE-CONVERTED parameter. Valid value YES/NO.	Program abends.	Correct problem and rerun job.
VIAG-0301-E	INTERNAL ERROR - GETMAIN FOR INPUT FILE DESCRIPTION FAILED Internal system error.	Program abends.	Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-0302-I	BRIDGE CONVERTING FILE <ddname> DCR <bridgeroutinename> File named in message is handled by Bridge.	None.	None.

Message	Description and Explanation	System Action	User Action
VIAG-0303-E	<ddname> DD STATEMENT MISSING DD named in message is missing. Bridge attempts to read information on a file it should handle, and the DD file is not found in JCL.	Program abends.	Add DD name or correct name and rerun job.
VIAG-0304-E	INTERNAL ERROR - RDJFCB FAILED, FILE <ddname>, RC: <returncode> Internal system error.	Program abends.	Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-0401-E	INVALID RECFM OF FILE <ddname> - MUST BE F/FB/FBA This file is unsupported.	Program abends.	Cannot be bridged.
VIAG-0402-E	INTERNAL ERROR - GETMAIN FOR INPUT BUFFER FAILED Internal system error.	Program abends.	Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-0404-E	ERROR IN TESTCB. DD:<ddname>. RC=<returncode>. REASON CODE=<reasoncode> Test CB command on VSAM file failed. See return and reason code.	Program abends.	Refer to return code and reason code in message. Correct problem and rerun job.
VIAG-0405-E	UNSUCCESSFUL OPEN. DD:<ddname>. RC=<returncode> Unsuccessful open to user file that should be handled by Bridge. See return code.	Program abends.	Refer to return code in message. Correct problem and rerun job.

Message	Description and Explanation	System Action	User Action
VIAG-0407-E	MISMATCH BETWEEN FILE AND DCR RKP. DD:<ddname>,DCR RKP(<rkp>), FILE RKP(<rkp>) Mismatch between file and DCR key definition. Bridge calculates new key position in record which is different from the one in the real VSAM file.	Program abends.	DCR probably not up to date. Regenerate DCR. If this does not work, the wrong VSAM file is being used.
VIAG-0451-E	FILE LRECL MISMATCH WAS DETECTED. DD :<ddname>. LRECL(<LRECL>), EXPECTED LRECL(<LRECL>) Bridge calculates expected file LRECL, which is not equal to real file LRECL.	Program abends.	DCR probably not up to date. Regenerate DCR. If this does not work, the file LRECL in the program did not change according to date definition.
VIAG-0452-E	RECFM MISMATCH WAS DETECTED. DD:<ddname> FILE-RECFM(<recfm>), DCR-RECFM(<recfm>) File record form mismatch. Record format in DCR id is different from record format of file in job.	Program abends.	DCR probably not up to date. Regenerate DCR. If this does not work, an incorrect file is being used in the job.
VIAG-0453-E	MISMATCH IN LRECL.DD:<ddname>,FILE-LRECL(<LRECL>) PROGRAM-LRECL(<LRECL>) File LRECL mismatch for variable record file, record length, or program record length in program. Delta is greater than the file record length due to date fields.	Program abends.	DCR probably not up to date. Regenerate DCR. If this does not work, an incorrect file is being used in SCL.
VIAG-0455-E	CALCULATED BLKSIZE (<blksize>) BIGGER THAN MAXIMUM BLKSIZE. FILE: <ddname>. File block size not defined in program or for the file, and Bridge calculation for blksize is greater than the maximum blksize.	Program abends.	Change file blocking factor so that blksize is less than 32K. Rerun job.

Message	Description and Explanation	System Action	User Action
VIAG-0501-E	INTERNAL ERROR - DDNAME:<ddname> NOT FOUND IN DDLIST Internal system error.	Program abends.	Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-0502-E	INTERNAL ERROR - GETMAIN FOR PROGRAM SAVE AREA FAILED Internal system error.	Program abends.	Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-0503-E	INTERNAL ERROR - CANNOT RESTORE AMDSB. DCR KEYS TABLE MISMATCH. DD:<ddname> Internal system error.	Program abends.	Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-0602-E	ERROR DETECTED DURING CONVERSION OF <ddname> INPUT RECORD, RC=<returncode> Cannot determine record type.	Program abends.	Examine data and make changes to Bridge Routine as required.
VIAG-0603-E	INVALID LRECL IN INPUT RECORD. DD: <ddname>, LENGTH:<lrecl> Variable record has invalid record length (less than 4) in RDW.	Program abends.	Review and correct input file. Rerun job.
VIAG-0613-E	DATE TRUNCATED WHEN CONVERTING FILE <ddname> Usually happens with variable length records when the date field is partially contained in the record.	Program abends.	DCR probably not up to date. Regenerate DCR.

Message	Description and Explanation	System Action	User Action
VIAG-0615-E	BAD DEPENDING ON VALUE WHEN CONVERTING FILE <ddname> Depending on variable contains invalid value.	Program abends.	Correct bad record.
VIAG-0616-E	INVALID RECORD TYPE WHEN CONVERTING FILE <ddname> Data in TYPE field not defined in DCR.	Program abends.	DCR probably not up to date. Regenerate DCR.
VIAG-0617-E	DATE TRUNCATED WHEN CONVERTING FIELD <fieldname> IN SEG <segmentname> of DBD <dbdname> Bridge determined that the field to convert is not as large as the Bridge Rule indicates and truncates the field.	Program abends.	Check Bridge Definition to make sure the correct Bridge Rule is assigned to field and that Bridge Rule is correctly defined.
VIAG-0619-E	INVALID RC <returncode> WHEN CONVERTING FILE <ddname> Internal system error.	Generate fails.	Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-0620-E	DATE DEFINED OUT OF AREA. FILE <ddname> Inconsistent date definition.	Program abends.	DCR probably not up to date. Regenerate DCR.
VIAG-0621-E	USER ERROR FROM CONVERSION MACRO Internal system error.	Program abends.	Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.

Message	Description and Explanation	System Action	User Action
VIAG-0701-E	INTERNAL ERROR - UNEXPECTED MACRF DETECTED IN DD:<ddname> Invalid MACRF parameter. Internal system error.	Program abends.	Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-0704-E	INTERNAL ERROR - UNEXPECTED PUT LOCATE DURING VSAM PUT, DD:<ddname> Internal system error.	Program abends.	Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-0705-E	UNEXPECTED LRECL (<lrecl>) RECEIVED DURING CONVERSION, DD:<ddname> Length of fixed output record after conversion is not equal to physical record length.	Program abends.	DCR probably not up to date. Regenerate DCR.
VIAG-0801-E	LOADING OF BRIDGE PARAMETERS MODULE FAILED Bridge general parameters module (VIAGBRGP) was not found or cannot be loaded.	Program abends.	Verify that Bridge load library is located in steplib on linklist. Rerun job.
VIAG-0802-E	DCR NAME <bridgeroutinename> NOT EXIST DCR named in message cannot be found.	Program abends.	Correct DCR name or add DCR library to steplib. Rerun job.
VIAG-0803-E	EXIT NAME <exitroutinename> NOT EXIST DCR user exit is not found or cannot be loaded. DCR user exit is specified in VIAGBRGP module.	Program abends.	Review user exit name and reason it cannot be loaded or remove exit name from VIAGBRGP module.
VIAG-0804-E	DCR NAME PARAMETER IS MISSING DCR name is not supplied to the DCR user exit module.	Program abends.	Correct problem and rerun job.

Message	Description and Explanation	System Action	User Action
VIAG-0805-E	PARM LENGTH IS TOO LONG DCR name length passed in JCL-PARM greater than 8 characters.	Program abends.	Correct DCR name parameter. Rerun job.
VIAG-0901-I	TOTAL RECORDS INPUT:000000000 Message from VIAGTEST.	N/A	N/A
VIAG-0902-I	STARTING AT RECORD..: 000000000 Message from VIAGTEST.	N/A	N/A
VIAG-0903-I	STOPPING AFTER RECORD…: 000000000 Message from VIAGTEST.	N/A	N/A
VIAG-0904-I	TOTAL RECORDS WRITTEN: 000000000 Message from VIAGTEST.	N/A	N/A

IMS Messages

Message	Description and Explanation	System Action	User Action
VIAG-2006-E	UNSUPPORTED IMS RELEASE Bridge does not support the IMS release.	Program abends.	
VIAG-2011-E	<modulename> CALLED <modulename> FROM POSITION <offset> AND RECEIVED R.C. <returncode> Internal system error.	Program abends.	Rerun program with trace. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-2012-I	<value1> "<value2>" Product internal information for problem determination.	N/A	N/A

Message	Description and Explanation	System Action	User Action
VIAG-2013-F	<pointername> IS A NULL POINTER AT OFFSET <offset> Internal system error.	Program abends.	Rerun program with trace. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-2014-F	<pointername> POINTS TO A NULL POINTER AT OFFSET <offset> Internal system error.	Program abends.	Rerun program with trace. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-2040-E	INTERNAL ERROR: (#1PRMADR-#1UPARMA) IS LT ZERO. #1UPARML = <value>, #1UPARAM =<value>, #0NEWPAD =<value> Internal system error.	Program abends.	Rerun program with trace. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-2041-E	INTERNAL ERROR: (#1PRMADR-#1UPARMA) IS GT #1UPARML. #1UPARML = <value>, #1UPARAM = <value>, #0NEWPAD = <value> Internal system error.	Program abends.	Rerun program with trace. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-2045-E	PROGRAM FAILED TO GET PCB INFO. RC = <returncode> Cannot locate a DCR for a PCB.	Program abends.	Verify that the missing DCR is in one of the DCR libraries used by the program (usually DCRLIB DD name). If the DCR does not exist because it is obsolete, regenerate the PCR. If the DCR is in one of the libraries, verify that it is loadable.

Message	Description and Explanation	System Action	User Action
VIAG-2046-E	INCONSISTENCY DETECTED: A SUCCESSFUL GET BUT GSAM FILE WAS NOT OPENED Internal system error.	Program abends.	Rerun program with trace. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-2047-E	A GET TYPE FUNCTION WITH IO AREA TOO SMALL. (SEE IMS ABEND CODE 200) Internal system error.	Program abends.	Rerun program with trace. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-2051-E	INCONSISTENCY DETECTED: QUALIFIED COMMAND BUT EMPTY HOLD TAB Internal system error.	Program abends.	Rerun program with trace. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-2052-E	INCONSISTENCY DETECTED: SEG NAME IN PCB MASK DOES NOT MATCH SELECTED SEG IN SSA AND STAT OK Internal system error.	Program abends.	Rerun program with trace. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-2053-E	INCONSISTENCY DETECTED: SEGMENT NAME RETURNED IN PCB MASK NOT FOUND IN PCR Internal system error.	Program abends.	Rerun program with trace. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.

Message	Description and Explanation	System Action	User Action
VIAG-2054-E	SEGMENT:<segmentname> IN PCB#:<pcbname> KEYLEN:<keylength> NOT = KEY FEEDBACK LEN:<value> Key length as found in PCBMASK is different than the one found in DCR.	Program abends.	DCR probably not up to date. Regenerate DCR.
VIAG-2055-E	KEY CONVERSION FAILED, SEGMENT NAME= <segmentname>, RC = <returncode> Bridge failed to convert concatenated key in PCBMASK.	Program abends.	This message is preceded by a message specifying reason for failure. Precede as recommended in previous message.
VIAG-2056-E	INCONSISTENCY DETECTED: SEG LEVEL IN PCBMASK NOT LOWER THAN LEVEL IN HOLD TAB. STAT: <value> Internal system error.	Program abends.	Rerun program with trace. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-2057-E	INTERNAL ERROR: REQUESTED ACTION IS NOT O OR N. #1PCMACT=<value> Internal system error.	Program abends.	Rerun program with trace. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-2061-E	INCONSISTENCY DETECTED: (#1CTYPE NOT#1CNVIN OR #1CNVOUT) OR #PPCBTYP NOT #PCBGSAM #1CTYPE = <value> #PPCBTYP = <value> Internal system error.	Program abends.	Rerun program with trace. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.

Message	Description and Explanation	System Action	User Action
VIAG-2062-E	INCONSISTENCY DETECTED: EMPTY HOLDTAB. #PHLDTAB = <value> #PHLDPTR = <value> Internal system error.	Program abends.	Rerun program with trace. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-2063-E	CONVERSION OF I/O AREA FAILED. IOATYPE = <value> #PHLDTAB = <value> INPUT ADDR IN R8 = <value> OUT AREA IN R4 = <value> System failed to convert I/O AREA.	Program abends.	This message is preceded by a message specifying reason for failure. Proceed as recommended in previous message.
VIAG-2064-E	INCONSISTENCY DETECTED: GSAM CONVERSION #1CTYPE IS NEITHER #1CNVIN NOR #1CNVOUT. #1CTYPE = <value> Internal system error.	Program abends.	Rerun program with trace. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-2065-E	INTERNAL ERROR: GSAM INPUT RECFM (#GRECFM) IS NOT F, V, OR U. #PGRECFM = <value> Internal system error.	Program abends.	Rerun program with trace. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-2066-E	INTERNAL ERROR: GSAM OUTPUT RECFM (#PGRECFM) IS NOT F, V, OR U. #PGRECFM = <value> Internal system error.	Program abends.	Rerun program with trace. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.

Message	Description and Explanation	System Action	User Action
VIAG-2067-E	GSAM CONVERSION ERROR CONV-GSAM RETURNED RC = 8 System failed to convert I/O AREA.	Program abends.	This message is preceded by a message specifying reason for failure. Proceed as recommended in previous message.
VIAG-2068-E	GSAM CONVERSION ERROR OF FIXED LENGTH RECORD. CONV-GSAM RETURNED RC = 4 I/O area length is not long enough to hold converted record.	Program abends.	DCR probably not up to date. Regenerate DCR.
VIAG-2069-E	INCONSISTENCY DETECTED: INPUT VARIABLE SEGMENT LENGTH EXCEEDS THE MAXIMUM SPECIFIED IN DCR Maximum segment length as held in DCR is not large enough to hold converted segment.	Program abends.	DCR probably not up to date. Regenerate DCR.
VIAG-2070-E	INTERNAL ERROR- DURING GSAM CONVERSION VARIABLE RECORD LENGTH LESS THAN ZERO Internal system error.	Program abends.	Rerun program with trace. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-2072-E	ERROR LOADING DCR EXIT: EXIM01 = <exitname> Cannot load DCR user exit routine.	Program abends.	Add PDS library containing the user exit routine to the STEPLIB.
VIAG-2076-E	DYNAMIC ALLOCATION ERROR ON PCRLIB R15= <value> REASON= <value> INFO= <value> DDNAME = <ddname> Dynamic allocation for the DCR library failed.	Program abends.	Verify that the default PCR library defined in VIAGBRGP is valid and available. Or explicitly specify the DCR library in the JCL (probably PCRLIB DD name).

Message	Description and Explanation	System Action	User Action
VIAG-2077-E	OPEN ERROR FOR PCRLIB R15=<value> DDNAME=<ddname> DSNAME = <dsname> System failed to open the PCR library.	Program abends.	Verify that the PCR library is a valid load library.
VIAG-2078-E	DYNAMIC ALLOCATION ERROR ON DCRLIB R15= <value> REASON= <value> INFO= <value> DDNAME =<ddname> Dynamic allocation for DCR library failed.	Program abends.	Verify that the default DCR library defined in VIAGBRGP is valid and available. Or explicitly specify the DCR library in the JCL (probably DLRLIB DD name).
VIAG-2079-E	OPEN ERROR FOR DCRLIB R15= <value> DDNAME= <ddname> DSNAME = <dsname> System failed to open the DCR library.	Program abends.	Verify that the DCR library is a valid library.
VIAG-2080-E	FREEMAIN ERROR FOR OLD PCB TABLES R15= <value> Internal system error.	Program abends.	Rerun program with trace. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-2081-E	GETMAIN ERROR FOR NEW PCB TABLES R15= <value> Failed to allocate memory for the PCB table.	Program abends.	Specify a larger region when running program.
VIAG-2082-E	UNSUCCESSFUL CALL TO USE-PCR #1NPSBNM = <value #1PCBLS# = <value> PCRU = <value>). Failed to load PCR.	Program abends.	Verify that the PCR is in one of the PCR load libraries available to the program. If it is, verify that it is loadable (PCRGEN successfully completed).

Message	Description and Explanation	System Action	User Action
VIAG-2083-E	NUMBER OF PCBS:<value> IN PSB (<psbname>) DIFFERENT FROM NUMBER IN PCR(<pcrname>) The number of the PCBS in the PSB is different than the number retained in the PCR.	Program abends.	PCR is probably not up to date. Regenerate PCR.
VIAG-2090-E	ADDR OF 1ST AND LAST BLOCK USED ARE INCONSISTENT. 0 AND NON-0 (<value1> <value2>). Internal system error.	Program abends.	Rerun program with trace. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-2091-E	GETMAIN FAILED. NO STORAGE AVAILABLE. RC = <value> No memory available for system blocks.	Program abends.	Rerun program with a larger region.
VIAG-2092-E	ADDRESS OF FIRST BLOCK IN CHAIN IS ZERO Internal system error.	Program abends.	Rerun program with trace. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-2093-E	ADDRESS OF LAST BLOCK USED IS ZERO Internal system error.	Program abends.	Rerun program with trace. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.

Message	Description and Explanation	System Action	User Action
VIAG-2094-E	INVALID REQUEST SIZE (<value>) GREATER THAN USABLE BLOCK SIZE (<value>) Internal system error.	Program abends.	Rerun program with trace. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-2101-E	INCONSISTENCY DETECTED: PCB MASK/NAME NOT FOUND IN PCB TAB Internal system error.	Program abends.	Rerun program with trace. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-2102-E	INCONSISTENCY DETECTED: REPL/DLET FUNC, BUT NO PREV HOLD REQUEST IN PCB: <pcbname> Internal system error.	Program abends.	Rerun program with trace. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-2103-E	INCONSISTENCY: IO-AREA NOT SUPPLIED IN PARMS FOR FUNC: <value> Internal system error.	Program abends.	Rerun program with trace. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-2104-E	INCONSISTENCY DETECTED: SEGMENT NAMEIN SSA NOT FOUND IN PCR (RE-GEN PCR) Segment specified in SSA cannot be located in the PCR.	Program abends.	PCR is probably not up to date. Regenerate PCR.
VIAG-2105-E	INCONSISTENCY: FIELD IN SSA NOT FOUND IN PCR (RE-GEN PCR) Field specified in SSA cannot be located in the PCR.	Program abends.	PCR is probably not up to date. Regenerate PCR.

Message	Description and Explanation	System Action	User Action
VIAG-2106-E	INCONSISTENCY DETECTED: UNQUALIFIED COMMAND WITH FUNC ISRT Internal system error.	Program abends.	Rerun program with trace. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-2107-E	INCONSISTENCY DETECTED: THE NUMBER OF PARAMETERS PASSED BY THE APPLICATION WAS ZERO BUT IMS COMMAND SUCCESSFULLY COMPLETED Internal system error.	Program abends.	Rerun program with trace. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-2108-E	INCONSISTENCY DETECTED: THE NUMBER OF PARAMETERS PASSED BY THE APPLICATION EXCEEDED THE MAXIMUM ALLOWED BUT IMS COMMAND SUCCESSFULLY COMPLETED Internal system error.	Program abends.	Rerun program with trace. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-2109-E	FI STATUS RETURN BY IMS WHEN SYSTEM SUPPLIED IO-AREA Internal system error.	Program abends.	Rerun program with trace. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-2110-E	INCONSISTENCY DETECTED: PCB TABLE IS EMPTY Internal system error.	Program abends.	Rerun program with trace. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.

Message	Description and Explanation	System Action	User Action
VIAG-2111-F	INCONSISTENCY DETECTED: RC - TABLE HAS WRONG PARAMETERS Internal system error.	Program abends.	Rerun program with trace. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-2112-F	INCONSISTENCY DETECTED: OUTPUT VARIABLE LENGTH SEGMENT EXCEEDS THE MAXIMUM SPECIFIED IN DCR AND IMS STATUS OK Internal system error.	Program abends.	Rerun program with trace. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-2113-F	INCONSISTENCY DETECTED: THE PRODUCT FAILED TO OPEN A GSAM FILE BUT ISRT SUCCESSFULLY COMPLETED Internal system error.	Program abends.	Rerun program with trace. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-2126-E	GET-AREA FROM <value> RETURNED RC = <value>. REQUESTED LENGTH = <value> System failed to allocate memory.	Program abends.	This message is preceded by a message specifying reason for failure. Proceed as recommended in previous message.
VIAG-2127-W	SEGMENT <segmentname> NOT FOUND IN PCR Segment specified in the SSA cannot be located in the DCR.	N/A	If DCR/PCR is up to date, IMS fails the request: fix application program. If DCR/PCR is not up to date and segment is defined in DCR: regenerate DCR/PCR.

Message	Description and Explanation	System Action	User Action
VIAG-2128-W	FIELD <fieldname> NOT FOUND IN PCR Field specified in SSA cannot be located in the DCR/PCR.	N/A	If DCR/PCR is up to date, IMS fails the request: fix application program. If DCR/PCR is not up to date: regenerate DCR/PCR.
VIAG-2129-E	CONV-FIELDS RETURNED RC <value> SSA conversion failed.	Program abends.	This message is preceded by a message specifying reason for failure. Precede as recommended in previous message.
VIAG-2130-E	TRIED TO ADD AN SSA FIELD TO A FULL CONVTAB Internal system error.	Program abends.	Rerun program with trace. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-2131-E	CONV-KEY RETURNED RC <value> SSA key conversion failed.	Program abends.	This message is preceded by a message specifying reason for failure. Precede as recommended in previous message.
VIAG-2140-I	REGISTERS AT ENTRY TO ABEND: Register, printout header.	N/A	N/A
VIAG-2141-I	GPR VALUES: Register, printout header.	N/A	N/A
VIAG-2142-I	<value1> <value2> <value3> <value4> Registers printout.	N/A	N/A
VIAG-2985-E	UNSUCCESSFUL LOAD (RC = <value>) OF PSB <psbname> PCRGEN could not allocate PSB.	Program abends.	Verify that PSB name is specified correctly and that it resides in a load library available to PCRGEN.

Message	Description and Explanation	System Action	User Action
VIAG-2986-E	RECEIVED PSBNAME LENGTH OF <length> GREATER THAN MAX OF LENGTH 8 Invalid PSB name specified for PCRGEN.	Program abends.	Correct PSB name.
VIAG-2995-E	UNSUCCESSFUL LOAD (RC = <value>) OF DBD <dbdname> DCRGEN could not locate DBD.	Program abends.	Verify that DBD name is spelled correctly and that it resides in a load library available to DCRGEN.
VIAG-2996-E	INVALID PARAMETER WAS SPECIFIED Invalid DBD name or DBDALIAS name specified for a logical DBD.	Program abends.	Correct names.
VIAG-2997-E	ACCESS METHOD <value> IS UNSUPPORTED Unsupported IMS access method.	Program abends.	N/A
VIAG-2998-E	INDEXED SEGMENT <segmentname> WITH TSGTAB ON (LCHCODE OF LCHILD) IS INVALID	Program abends.	Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-2999-E	INDEXED SEGMENT <segmentname> HAS NO SRCH FIELDS DEFINED IN INDXTAB Unexpected or unsupported DBD internal structure.	Program abends.	Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-3125-E	CANNOT LOCATE FIELD <fieldname> FOR KEY <keyname> OF RECORD <recordname> Internal system error.	Generate fails.	Add SUPPRT= parameter and rerun GEN. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.

Message	Description and Explanation	System Action	User Action
VIAG-3126-W	TWO DEPENDING ON ARRAYS FOR FIELD <fieldname> OF SEGMENT/RECORD <segment/recordname>	Program may abend at runtime	Specify only 1 array for a particular index variable.
VIAG-3127-W	FIELD <fieldname> OUTSIDE OF RECORD/SEGMENT <recordname/segmentname> Or FIELD <fieldname> HAS I/O FLD OUTSIDE OF DB FLD IN RECORD <recordname>	Program may abend at runtime.	A related field (input/output field) must be defined in the same database field as the field undergoing conversion. If the related field is outside the database field, an abend may occur when access to the related field is attempted.
VIAG-3128-E	CANNOT CONVERT DATA OF X TYPE FIELD <fieldname> IN SEGMENT <segmentname> Internal system error.	Generate fails.	Add SUPPRT= parameter and rerun GEN. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-3129-E	BAD OR MISSING PARAMETER DEFINITION Internal system error.	Generate fails.	Capture the generate JCL and contact the ASG Service Desk.
VIAG-3130-E	UNEXPECTED END OF FILE IN VIAGCNTO/VIAGCNTN Internal system error.	Generate fails.	Add SUPPRT= parameter and rerun GEN. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.

Message	Description and Explanation	System Action	User Action
VIAG-3131-E	INVALID RECORD IN VIAGCNTO/VIAGCNTN Internal system error.	Generate fails.	Add SUPPRT= parameter and rerun GEN. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-3132-E	MISSING <name> RECORD IN DATABASE EXTRACTED AREA The named label was not found in the database extracted area.	Generate fails.	Add SUPPRT= parameter and rerun GEN. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-3133-E	MISMATCH BETWEEN OLD AND NEW DATABASE DEFINITIONS The old and new database definitions have differences which are not accounted for solely by field conversion rules.	Generate fails.	Correct the errors and rerun GEN.
VIAG-3201-E	INVALID PCBID <value> FOR PCR <pcrname> Internal system error.	Program abends.	Rerun program with trace. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-3202-E	CANNOT LOCATE FIELD <fieldname> SEGMENT <segmentname> OF PCB NUMBER <value> OF PCR <pcrname> Internal system error.	Program abends.	Rerun program with trace. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.

Message	Description and Explanation	System Action	User Action
VIAG-3203-E	CANNOT LOCATE SEGMENT <segmentname> OF PCB <value> IN PCR <pcrname> Cannot locate a key segment. Error can occur when a DCR is not up to date.	Program abends.	DCR probably not up to date. Regenerate DCR. Then, if the problem reoccurs: Rerun program with trace. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-3204-E	<value> OF PCB <value> IN PCR <pcrname> NOT FOUND IN DCR <dcrname> Cannot locate a segment in DCR.	Program abends if segment located by IMS. Otherwise, IMS error returned.	If IMS error returned, correct application error. Otherwise: DCR probably not up to date. Regenerate DCR. **OR** PCR is probably not up to date. Regenerate PCR.
VIAG-3205-E	CANNOT ALLOCATE STORAGE FOR <value> SIZE <value> Cannot allocate necessary memory.	Program abends.	Rerun program with larger region.
VIAG-3206-E	CANNOT FIND <value> <value> DCR/PCR cannot be loaded.	Program abends.	Verify that required DCR/PCR resides in DCR/PCR load libraries available to the system (probably DCRLIB/PCRLIB DD name).
VIAG-3207-E	BAD PCB ID <value>. PCR <pcrname> HAS <value> PCBS DEFINED Internal system error.	Program abends.	Rerun program with trace. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.

Message	Description and Explanation	System Action	User Action
VIAG-3213-E	<value>: DATE TRUNCATED WHEN CONVERTING SEGMENT <segmentname> OF DBD <dbdname> A date is partially contained in an IMS segment. Usually happen with variable length segments when date field is partially contained in segment.	Program abends.	Correct user error and rerun GEN.
VIAG-3214-E	<value>: BAD DEP ON VALUE WHEN CONVERTING FIELD <fieldname> IN SEG <segmentname> OF DBD <dbdname> The depending on variable contains an invalid value.	Program abends.	Correct bad record.
VIAG-3215-E	<value>: BAD DEP ON VALUE WHEN CONVERTING SEGMENT <segmentname> OF DBD <dbdname> The depending on value contains an invalid value.	Program abends.	Correct bad record.
VIAG-3216-E	<value>: INVALID TYPE WHEN CONVERTING SEGMENT <segmentname> OF DBD <dbdname> Internal system error.	Program abends.	Rerun program with trace. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-3217-E	<value>: DATE TRUNCATED WHEN CONVERTING FIELD <fieldname> IN SEG <segmentname> OF DBD <dbdname> A date field is partially contained in an IMS field. This field cannot be converted although the whole field can.	Program abends.	Correct user error and rerun GEN.

Message	Description and Explanation	System Action	User Action
VIAG-3218-E	<value>: INVALID RC <value> WHEN CONVERTING FIELD <fieldname> IN SEGMENT <segmentname> OF DBD <dbdname> Internal system error.	Program abends.	Rerun program with trace. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-3219-E	<value>: INVALID RC <value> WHEN CONVERTING SEGMENT <segmentname> OF DBD <dbdname> Internal system error.	Program abends.	Rerun program with trace. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-3220-E	<value>: DATE DEFINED OUT OF SEGMENT <segmentname> OF DBD <dbdname> Inconsistent date definition.	Program abends.	DCR probably not up to date. Regenerate DCR.
VIAG-3221-E	<value>: USER ERROR FROM CONVERSION MACRO User error from conversion rule.	Program abends.	Rerun program with trace. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-3222-E	BLDL FAILED. RC=<value> Internal system error.	Program abends.	Rerun program with trace. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-3223-E	GETMAIN FAILED. RC=<value> Cannot allocate memory for system.	Program abends.	Rerun program with larger region.

Message	Description and Explanation	System Action	User Action
VIAG-3224-E	CANNOT LOAD DCR <dcrname> FOR LOGICAL SEGMENT <segmentname> OF PCB <pcbname> Cannot load DCR of logical segment.	Program abends.	Verify that physical DCR is in a DCR load library available to system.
VIAG-3225-E	BAD <value> <value> LOADED AT ADDRESS <value>. NAME IN LOADED <value> IS <value>	Program abends.	Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-3226-E	LOADED DCR <dcrname> IS OF TYPE <value>. THIS TYPE IS INVALID FOR IMS	Program abends.	Make sure DCR is defined correctly.

Bridge Generation Messages

These are the Bridge generation messages:

Message	Description and Explanation	System Action	User Action
VIAG-3101-E	INVALID DATESDEF FILE Internal system error.	Generate fails.	Add SUPPRT= parameter and rerun GEN. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-3102-E	INVALID DATE DEFINITION STATEMENT Internal system error.	Generate fails.	Add SUPPRT= parameter and rerun GEN. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-3103-E	NO DATES DEFINITION IN FILE Internal system error.	Generate fails.	Add SUPPRT= parameter and rerun GEN. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-3104-E	INVALID LINE FOUND Internal system error.	Generate fails.	Add SUPPRT= parameter and rerun GEN. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.

Message	Description and Explanation	System Action	User Action
VIAG-3105-E	INVALID DATE TYPE/OFFSET RECORD Internal system error.	Generate fails.	Add SUPPRT= parameter and rerun GEN. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-3106-E	INCOMPATIBLE DATE DEFINITIONS Internal system error.	Generate fails.	Add SUPPRT= parameter and rerun GEN. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-3107-E	CONDITION NOT ALLOWED AFTER DEFAULT STATEMENT Internal system error.	Generate fails.	Add SUPPRT= parameter and rerun GEN. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-3108-E	EXTRANEOUS RECORD Internal system error.	Generate fails.	Add SUPPRT= parameter and rerun GEN. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.

Message	Description and Explanation	System Action	User Action
VIAG-3109-E	INVALID OR MISSING INIT RECORD Internal system error.	Generate fails.	Add SUPPRT= parameter and rerun GEN. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-3110-E	INCOMPLETE SOURCE Internal system error.	Generate fails.	Add SUPPRT= parameter and rerun GEN. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-3111-E	TOO MANY VECTOR DEFINITIONS FOR ONE RECORD TYPE Internal system error.	Generate fails.	Add SUPPRT= parameter and rerun GEN. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-3112-E	SPECIFIED DATE/PAD FIELDS IN DCR OVERLAP Two date fields or padding fields fully or partially overlap.	Generate fails.	Correct user error and rerun GEN.
VIAG-3113-W	PADDING FIELD OVERLAPS BEGINNING OF FIELD <fieldname> IN SEGMENT <segmentname> Date field or padding field partially contained in an IMS field.	Program abends if the field is used during runtime.	Correct the Bridge definition if the field is used during runtime.

Message	Description and Explanation	System Action	User Action
VIAG-3114-W	DATE OVERLAPS BEGINNING OF FIELD <fieldname> IN SEGMENT <segmentname> Date field or padding field is partially contained in an IMS field.	Program abends if the field is used during runtime.	Correct the Bridge definition if the field is used during runtime.
VIAG-3115-W	PADDING FIELD OVERLAPS END OF FIELD <fieldname> IN SEGMENT <segmentname> Date field or padding field partially contained in an IMS field.	Program abends if the field is used during runtime.	Correct the Bridge definition if the field is used during runtime.
VIAG-3116-W	DATE OVERLAPS END OF FIELD <fieldname> IN SEGMENT <segmentname> Date field or padding field partially contained in an IMS field.	Program abends if the field is used during runtime.	Correct the Bridge definition if the field is used during runtime.
VIAG-3117-E	BAD LEN FOR FIELD <fieldname> IN SEG <segmentname>. OLD LEN <value>, NEW LEN <value>, EXPECTED <value> Lengths of field in old DBD and converted DBD do not correspond to dates defined for the field.	Generate fails.	Correct user error and rerun GEN.
VIAG-3118-W	ARRAY OVERLAPS START OF FIELD <fieldname> IN SEGMENT <segmentname> Array partially contained in an IMS field.	Program abends if the field is used during runtime.	Correct the Bridge definition if the field is used during runtime.
VIAG-3119-W	ARRAY OVERLAPS END OF FIELD <fieldname> IN SEGMENT <segmentname> Array partially contained in an IMS field.	Program abends if the field is used during runtime.	Correct the Bridge definition if the field is used during runtime.

Message	Description and Explanation	System Action	User Action
VIAG-3120-W	DEPENDING ON VARIABLE NOT FULLY CONTAINED IN FIELD <fieldname> OF SEGMENT <segmentname> Depending on field partially contained in an IMS field.	Program abends if the field is used during runtime.	Correct the Bridge definition if the field is used during runtime.
VIAG-3121-E	ARRAY OVERLAPS DATE BOUNDARIES Date field partially contained in an array.	Generate fails.	Correct user error and rerun GEN.
VIAG-3122-E	RECFM FIXED SPECIFIED, BUT DATA HAS VARIABLE CHARACTERISTICS RECFM fixed was specified although record has variable characteristics such as occurs depending on.	Generate fails.	Correct user error and rerun GEN.
VIAG-3123-E	KEY CONTAINS DATE FIELD VSAM MREY contains date. This is not supported in the current Bridge release.	Generate fails.	Correct user error and rerun GEN.
VIAG-3124-E	DIFFERENT KEY POSITIONS IN DIFFERENT RECORD TYPES Conversion of different record types caused the key to move to different positions in the record for the different record types. Results from data preceding the key having different conversion rules for different record types that convert to different new sizes.	Generate fails.	Correct user error and rerun GEN. Padding fields can be used to ensure that the key is in the same position for all record types.
VIAG-3150-E	SEGMENT <segmentname> NOT FOUND IN DBD Segment not found in DBD.	Generate fails.	Segment is probably specified incorrectly. Correct user error and rerun GEN.

Message	Description and Explanation	System Action	User Action
VIAG-3183-E	SEGMENT <segmentname> IN LOGICAL DBD NOT FOUND IN PHYSICAL DBD <dbdname> Cannot locate segment defined in a logical DBD.	Generate fails.	The physical DCR is probably not up to date. Regenerate it and then generate the logical DCR.
VIAG-3184-E	CANNOT LOCATE REAL LOG CHILD <value> FOR VIRT LOG CHILD <value> Cannot locate the physical segment of a virtual logical child.	Generate fails.	The physical DCR is probably not up to date. Regenerate it and then generate the logical DCR.
VIAG-3185-E	CANNOT LOCATE DCR <dcrname> FOR VIRTUAL LOGICAL CHILD <segmentname> Cannot locate DCR for virtual logical child.	Generate fails.	The physical DCR is probably not up to date. Regenerate it and then generate the logical DCR.
VIAG-3186-E	PHYSICAL DCR <dcrname> FOR LOGICAL SEGMENT <segmentname> HAS AN INCONSISTENT CONV TYPE <value> A logical DCR and all of the physical DCRs is accesses must have the same conversion type: either file converted yes or file converted no. This is not true for this DCR.	Generate fails.	Correct user error and rerun GEN.
VIAG-3187-E	CANNOT LOCATE DCR <dcrname> FOR LOGICAL SEGMENT <segmentname> Cannot locate DCR for virtual logical segment.	Generate fails.	The physical DCR is probably not up to date. Regenerate it and then generate the logical DCR.
VIAG-3188-E	SEGMENT <segmentname> IN NEW DBD NOT FOUND IN OLD DBD The old DBD and the converted DBD are not the same. The only differences allowed by the program are those caused by field conversion rules.	Generate fails.	Correct user error and rerun GEN.

Message	Description and Explanation	System Action	User Action
VIAG-3189-E	DIFFERENT <value> IN OLD DBD (<dbdname>) AND NEW DBD (<dbdname>) IN SEGMENT <segmentname> The old DBD and the converted DBD are not the same. The only differences allowed by the program are those caused by field conversion rules.	Generate fails.	Correct user error and rerun GEN.
VIAG-3190-E	DIFFERENT <value> IN OLD DBD (<dbdname>) AND NEW DBD (<dbdname>) The old DBD and the converted DBD are not the same. The only differences allowed by the program are those caused by field conversion rules.	Generate fails.	Correct user error and rerun GEN.
VIAG-3191-E	BAD SIZE FOR SEGMENT <segmentname>. OLD SIZE <value> NEW SIZE <value> EXPECTED <value> The old DBD and the converted DBD are not the same. The only differences allowed by the program are those caused by field conversion rules.	Generate fails.	Correct user error and rerun GEN.
VIAG-3193-E	FIELD <fieldname> IN SEGMENT <segmentname> IS DEFINED DIFFERENTLY IN THE OLD DBD AND IN THE NEW ONE The old DBD and the converted DBD are not the same. The only differences allowed by the program are those caused by field conversion rules.	Generate fails.	Correct user error and rerun GEN.

Message	Description and Explanation	System Action	User Action
VIAG-3194-E	FIELD <fieldname> IN NEW DBD SEGMENT <segmentname> NOT FOUND IN OLD DBD The old DBD and the converted DBD are not the same. The only differences allowed by the program are those caused by field conversion rules.	Generate fails.	Correct user error and rerun GEN.
VIAG-3195-E	BAD FORMAT IN NEW DBD FILE Internal system error.	Generate fails.	Add SUPPRT= parameter and rerun GEN. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-3196-E	BAD FORMAT IN OLD DBD FILE Internal system error.	Generate fails.	Add SUPPRT= parameter and rerun GEN. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-3197-E	OLD DBD NAME <dbdname> AND NEW DBDNAME <dbdname> ARE NOT THE SAME The old DBD and the converted DBD are not the same. The only differences allowed by the program are those caused by field conversion rules.	Generate fails.	Correct user error and rerun GEN.

Message	Description and Explanation	System Action	User Action
VIAG-3198-E	NO RECORDS FOUND IN NEW DBD Internal system error.	Generate fails.	Add SUPPRT= parameter and rerun GEN. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-3199-E	NO RECORDS FOUND IN OLD DBD Internal system error.	Generate fails.	Add SUPPRT= parameter and rerun GEN. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-3301-E	CANNOT FIND/LOAD DCR <dcrname> Cannot locate physical in DCRs.	Generate fails.	Make sure correct DCR libraries are available.
VIAG-3302-E	CANNOT LOCATE SEGMENT <segmentname> FOR PCB NUMBER <value> Cannot locate segment in DCR.	Generate fails.	DCR probably not up to date. Regenerate DCR.
VIAG-3303-E	CANNOT LOCATE FIELD <fieldname> IN SEGMENT <segmentname> FOR PCB NUMBER <value> Cannot locate field in DCR.	Generate fails.	DCR probably not up to date. Regenerate DCR.
VIAG-3304-E	PCB DEFINED KEY LENGTH <value> SHORTER THAN EXPECTED <value> FOR PCB NO. <value> Key length in PSB is not large enough.	Generate fails.	Modify PSB key length and regenerate the old and converted PSBs and the PCR.

Message	Description and Explanation	System Action	User Action
VIAG-3305-E	INTERNAL ERROR: <value> FOR <value> PSB EXTRACTED DATA Internal system error.	Generate fails.	Add SUPPRT= parameter and rerun GEN. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-3306-E	NEW PSB DIFFERS FROM OLD ONE The old PBS and the converted PBS are not the same. The only differences allowed by the program are those caused by conversion rules.	Generate fails.	Correct user error and rerun GEN.
VIAG-3307-E	NEW PCB(<pcbname>) DIFFERS FROM OLD ONE The old PBS and the converted PBS are not the same. The only differences allowed by the program are those caused by conversion rules.	Generate fails.	Correct user error and rerun GEN.
VIAG-3308-E	NEW SEGMENT <segmentname> OF PCB NO. <value> DIFFERS FROM OLD ONE The old PBS and the converted PBS are not the same. The only differences allowed by the program are those caused by conversion rules.	Generate fails.	Correct user error and rerun GEN.
VIAG-3309-E	NEW FIELD <fieldname> DETECTED IN SEGMENT <segmentname> OF PCB NO. <value> The old PBS and the converted PBS are not the same. The only differences allowed by the program are those caused by conversion rules.	Generate fails.	Correct user error and rerun GEN.

Message	Description and Explanation	System Action	User Action
VIAG-3310-E	CANNOT LOCATE KEY OF DBD <dbdname> IN SEGMENT <segmentname> FOR PCB NUMBER <value> Internal system error.	Generate fails.	Add SUPPRT= parameter and rerun GEN. Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-3311-E	CANNOT LOCATE <value> <value> FOR LOGICAL SEGMENT <segmentname> OF PCB NUMBER <value> Cannot locate element for logical segment.	Generate fails.	DCR probably not up to date. Regenerate DCR.

CICS Messages

These are the CICS messages:

Note:

The transaction abend code used for all E-level messages is B2KU.

Message	Description and Explanation	System Action	User Action
VIAG-3500-I	TRANSACTION <*tranid*> PROGRAM <*programid*> TERM <*termid*> SEE FOLLOWING MSG: <*messagenumber*> General message printing tran ID, task number, program name, terminal name, and following message ID.	N/A	N/A
VIAG-3501-E	LOAD OF VIAGFCIC STUB VIAGCSTB FAILED - CALL DC ADMIN. Load of stub module failed.	Program abends.	Determine why the stub module cannot be loaded. Correct the problem and rerun program.
VIAG-3502-E	BRIDGE TASK RELATED USER EXIT NOT ACTIVATED PROPERLY - CALL DC ADMIN. Bridge not initiated or not activated properly.	Program abends.	Activate Bridge properly. If problem reoccurs: Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-3511-E	BAD GLOBAL AREA IN START PROGRAM (O2KDCSTR) - CALL DC ADMIN. Internal system error.	Bridge not initiated.	Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.

Message	Description and Explanation	System Action	User Action
VIAG-3512-E	LOAD OF VIAGSMSG (MESSAGE TABLE) FAILED - CALL DC ADMIN. Message module load failed.	Bridge not initiated.	Determine why the message module cannot be loaded (not defined, disabled, etc.). Correct the problem and restart Bridge.
VIAG-3513-E	LINK TO VIAGCSTP (STOP PROGRAM) FAILED - CALL DC ADMIN. Internal system error.	Program abends.	Determine why the link failed for this module. Correct the problem and restart Bridge.
VIAG-3514-E	INITIALIZE FAILED. REPLY GO OR CANCEL (SHUTDOWN IMMEDIATE CICS) Initialization of Bridge failed during startup of CICS. Bridge stopped and waited for reply from operator.	Message with reply to operator.	Reply "go" to continue or "cancel" to shutdown immediately.
VIAG-3515-W	RECEIVED GO AFTER FAILED - BRIDGE NOT INITIALIZED Bridge initialization failed and operator replied "go" to previous message.	Bridge not initialized during CICS startup.	Determine what caused the problem and why Bridge was not initialized. Correct the problem and restart Bridge.
VIAG-3516-W	RECEIVED CANCEL AFTER FAILED - CICS WILL BE SHUTDOWN IMMEDIATELY Bridge initialize failed and operator replied "cancel" to previous message.	Bridge not initiated. CICS shuts down immediately.	Determine what caused the problem and why Bridge was not initialized. Correct the problem and restart Bridge.
VIAG-3517-I	BRIDGE INITIALIZED AND SET TO ON Bridge successfully initialized.	N/A	N/A

Message	Description and Explanation	System Action	User Action
VIAG-3518-E	BRIDGE ALREADY INITIALIZED Received request to initialize Bridge after already initialized.	Ignore the request.	N/A
VIAG-3519-E	LOADING OF BRIDGE PARAMETERS MODULE (VIAGBRGP) FAILED Parameter module cannot be loaded.	Bridge is not initialized.	Determine why the parameter module cannot be loaded (not defined, disabled, etc.). Correct the problem and restart Bridge.
VIAG-3520-E	ENABLE OF EXIT <exitname> PROGRAM <programname> FAILED Problem enabling the exit Bridge is using.	Bridge not initiated.	Determine why enable the exit failed. Correct the problem and try to reactivate Bridge.
VIAG-3541-I	BRIDGE SET TO OFF SUCCESSFULLY COMPLETED Bridge successfully deactivated (set to off).	N/A	N/A
VIAG-3600-E	INVALID REQUEST TO TASK-RELATED USER EXIT Internal system error.	Program abends.	Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-3601-E	BAD GLOBAL AREA IN TASK-RELATED USER EXIT (GALENGTH AREA) Internal system error.	Program abends.	Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.

Message	Description and Explanation	System Action	User Action
VIAG-3602-E	BAD LOCAL AREA IN TASK-RELATED USER EXIT (TALENGTH AREA) Internal system error.	Program abends.	Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-3603-E	BAD PROGRAM BLOCK (INTERNAL ERROR) Internal system error.	Program abends.	Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-3604-E	BAD FILE BLOCK (INTERNAL ERROR) Internal system error.	Program abends.	Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-3605-E	BAD USER BLOCK (INTERNAL ERROR) Internal system error.	Program abends.	Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-3606-E	BAD PARAMETERS SUPPLIED TO BRIDGE (ZERO OR EVEN) Application program called the Bridge initialization routine with no parameters or with even parameters.	Program abends.	Correct the problem by calling the INIT routine properly and rerun the program.
VIAG-3607-E	PROBLEM IN LOADING DATE CONVERSION ROUTINE (DCR). DCR NAME<bridgeroutinename> Bridge cannot load the DCR.	Program abends.	Determine why Bridge cannot load the DCR. Correct the problem and restart Bridge.

Message	Description and Explanation	System Action	User Action
VIAG-3608-E	SAME FILE SUPPLIED BUT DCR NOT THE SAME. FILENAME: <filename> DCR1: <bridgeroutinename> DCR2: <bridgeroutinename> Application program called the INIT routine with the same file name but with a different DCR name.	Program abends.	Correct the problem by calling the INIT routine properly and rerun the program.
VIAG-3609-E	NUMBER OF FILES IS OVER MAXIMUM FOR 1 PROGRAM (<numberoffiles) Program exceeds the limit of files that can be handled by Bridge for one program.	Program abends.	Correct the program or report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-3610-E	BAD FILE NAME OR UNABLE TO ACCESS FILE. FILENAME: <filename> Application program called the INIT routine with a bad file name or the file cannot be accessed by Bridge.	Program abends.	Determine why the file cannot be accessed. Correct the problem and restart Bridge.
VIAG-3611-E	TYPE OF FILE NOT SUPPORTED. FILENAME: <filename> Bridge supports only VSAM files. Application program called the INIT routine with a file name that is not VSAM.	Program abends.	Correct the problem and restart Bridge.
VIAG-3612-E	BAD USER EXIT OR UNABLE TO LINK TO THIS EXIT. EXIT NAME: <exitname> INIT routine cannot link to the user exit routine mentioned in the parameter module.	Program abends.	Determine why link failed. Correct the problem and restart Bridge.

Message	Description and Explanation	System Action	User Action
VIAG-3613-E	BAD RECORD FORMAT OR RECORD LENGTH. FILENAME: <filename> DCR NAME: <bridgeroutinename> Mismatch between the DCR and the file details in CICS.	Program abends.	Correct the DCR and rerun the program.
VIAG-3614-E	NO RECORD FORMAT IN DCR FOR REMOTE FILE. FILENAME: <filename> DCR NAME: <bridgeroutinename> Mismatch between the DCR and file details in CICS.	Program abends.	Correct the DCR and rerun the program.
VIAG-3615-E	RECFM IS FIXED, BUT DATA HAS VARIABLE CHARACTERISTICS. FILENAME: <filename> DCR: <bridgeroutinename> Mismatch between the DCR and file details in CICS.	Program abends.	Correct the DCR and rerun the program.
VIAG-3616-E	BAD BRIDGE TYPE. MUST BE NORMAL OR REVERSE BRIDGING Internal system error.	Program abends.	Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-3617-E	PROGRAM SHOULD USE ONLY ONE TYPE OF BRIDGING (NORMAL OR REVERSE) Application program called the normal bridging routine and the reverse bridging routine in the same program.	Program abends.	Correct the problem and restart Bridge.

Message	Description and Explanation	System Action	User Action
VIAG-3618-E	FILE CAN BE USE ONLY WITH ONE TYPE OF BRIDGING. FILENAME: <filename> Application program called the normal bridging routine and the reverse bridging routine for the same file. File can be used with only one type.	Program abends.	Correct the problem and restart Bridge.
VIAG-3700-E	FILE CONTROL EXIT (XFCREQ) NOT ACTIVATED PROPERLY - CALL DC ADMIN. Internal system error.	Program abends.	Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-3701-E	BAD GLOBAL AREA IN FILE CONTROL EXIT (XFCREQ) - CALL DC ADMIN. Internal system error.	Program abends.	Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-3702-E	BAD TASK BLOCK IN FILE CONTROL EXIT (XFCREQ) - CALL DC ADMIN. Internal system error.	Program abends.	Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-3703-E	BAD USER BLOCK IN FILE CONTROL EXIT (XFCREQ) - CALL DC ADMIN. Internal system error.	Program abends.	Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.

Message	Description and Explanation	System Action	User Action
VIAG-3704-E	BAD RETURN CODE FROM DCR - DCRNAME: <bridgeroutinename> RETURN CODE: <returncode> Bad return code from conversion routine. This list provides RC numbers and their corresponding message IDs for this RC (format RC, ID; RC, ID, etc.): 4, 620; 7, 621; 8, 613; 9, 615; 10, 616; 11, 617; other, 619.	Program abends.	Correct DCR, correct program, or report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-3800-E	BAD ACTIVATE OF FILE CONTROL EXIT (XFCREQC) - CALL DC ADMIN. Internal system error.	Program abends.	Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-3801-E	BAD GLOBAL AREA IN FILE CONTROL EXIT (XFCREQC) - CALL DC ADMIN. Internal system error.	Program abends.	Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-3802-E	BAD TASK BLOCK IN FILE CONTROL EXIT (XFCREQC) - CALL DC ADMIN. Internal system error.	Program abends.	Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-3803-E	BAD USER BLOCK IN FILE CONTROL EXIT (XFCREQC) - CALL DC ADMIN. Internal system error.	Program abends.	Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.

Message	Description and Explanation	System Action	User Action
VIAG-3804-E	BAD RETURN CODE FROM DCR - DCRNAME: <bridgeroutinemane> RETURN CODE: <returncode> Bad return code from conversion routine. This list provides RC numbers and their corresponding message IDs for this RC (format: RC, ID; RC, ID, etc.): 4, 620; 7, 621; 8, 613; 9, 615; 10, 616; 11, 617; other, 619.	Program abends.	Correct DCR or report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-3805-E	NO LENGTH IN COMMAND - INTERNAL ERROR (XFCREQC) - CALL DC ADMIN. Internal system error.	Program abends.	Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-3806-E	BAD RECORD FORMAT OR RECORD LENGTH IN DCR. FILENAME: <filename> DCR NAME: <bridgeroutinename> Mismatch between DCR and file details in CICS.	Program abends.	Correct DCR and rerun program.
VIAG-4092-E	ERROR IN PARM - NO COMMA	Program abends.	Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-4093-E	ERROR IN PARM - PARM LENGTH IS TOO SHORT	Program abends.	Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.

Message	Description and Explanation	System Action	User Action
VIAG-4094-E	ERROR IN PARM - START PARM MUST BE 1 OR 2	Program abends.	Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.
VIAG-4095-E	PARM NOT FOUND - <parmname>	Program abends.	Report problem and all relevant information (including trace option, if available) to the ASG Service Desk.

CICS DL/I Messages

These are the CICS DL/I messages:

Message	Description and Explanation	System Action	User Action
VIAG-3900-I	TRANSACTION <value> (<value>) TERM <value> PSB NAME <psbname> SEE FOLLOWING MSG: <value>	N/A	N/A
VIAG-3910-E	BAD GLOBAL AREA IN DLI EXIT (XDLIPRE). CALL DC ADMIN. Internal system error.	Program abends.	Report problem and all relevant information (including trace output, if available) to the ASG Service Desk.
VIAG-3911-E	BAD TASK BLOCK IN DLI EXIT (XDLIPRE). CALL DC ADMIN. Internal system error.	Program abends.	Report problem and all relevant information (including trace output, if available) to the ASG Service Desk.

Message	Description and Explanation	System Action	User Action
VIAG-3912-E	BAD USER BLOCK IN DLI EXIT (XDLIPRE). CALL DC ADMIN. Internal system error.	Program abends.	Report problem and all relevant information (including trace output, if available) to the ASG Service Desk.
VIAG-3930-E	PSB ACTIVE - TERM FUNC EXPECTED BEFORE CALL TO INIT. Application program issues a call to the INIT routine while PSB is active. The call to the INIT routine must be performed before schedule command or after term command.	Program abends.	Correct the program and rerun it.
VIAG-3950-E	GSAM NOT SUPPORTED IN CICS ENVIRONMENT Internal system error.	Program abends.	Report problem and all relevant information (including trace output, if available) to the ASG Service Desk.
VIAG-3955-E	SCHD FUNC EXPECTED AFTER CALL TO INIT BRIDGE BUT OTHER FUNCTION SUPPLIED The application program must issue schedule command after call to the INIT routine. Another command was given instead of the schedule command.	Program abends.	Correct the program and rerun it.
VIAG-3956-E	MISMATCH BETWEEN INIT PSB AND SCHD PSB Mismatch between name of the PCR (and PCB name in the PCR) and PSB name in the schedule command.	Program abends.	Correct the program and verify that the INIT call (PSR name) and the schedule command are matched.

Message	Description and Explanation	System Action	User Action
VIAG-3960-E	#1CHCKRC = <value>, R.C. IS NOT OK Internal system error.	Program abends.	Report problem and all relevant information (including trace output, if available) to the ASG Service Desk.
VIAG-3961-E	NUMBER OF PCBS:<value> IN PSB (<psbname>) DIFFERENT FROM NUMBER IN PCR(<pcrname>), <VALUE> Mismatch between number of PCBs in PCR supplied by application program and the number of PCBs after schedule command.	Program abends.	Correct the program or the PCR and rerun the program.

IDMS Messages

These are the IDMS messages:

Message	Description and Explanation	System Action	User Action
VIAG-1100-E	… OF SUBSCHEMA <subschemaname> OF SCR <scrname> WHICH HAS NO DCR Continuation of another error message, providing additional information.	Program Abends	Report Problem and all relevant information (including trace output, if available) to the ASG Service Desk.
VIAG-1101-E	INVALID SSCID <sscid> for SCR <scrname> The SSCID is invalid for the SCR named in the message.	Program Abends	Report Problem and all relevant information (including trace output, if available) to the ASG Service Desk.

Message	Description and Explanation	System Action	User Action
VIAG-1102-E	CANNOT LOCATE RECORD <recordname> OF SUBSCHEMA <subschemaname> IN SCR <scrname> The program specified a record name which is not listed in the SCR.	Program Abends	Ensure that the Bridge Definition includes this record name, and regenerate the DCR and the SCRs that refer to it.
VIAG-1103-E	… IN SUBSCHEMA <subschemaname> OF SCR <scrname> Continuation of another error message, providing additional information.	Program Abends	N/A.
VIAG-1104-E	RECORD <recordname> of SUBSCHEMA <subschemaname> IN SCR <scrname> NOT FOUND IN DCR <dcrname>. The program specified a record name which is listed in the SCR but which is not listed in the DCR.	Program Abends	Ensure that the Bridge Definition includes this record name, and regenerate the DCR and the SCRs that refer to it.
VIAG-1105-E	CANNOT CONVERT RECORD <recordname> … Internal Error	Program Abends	Report Problem and all relevant information (including trace output, if available) to the ASG Service Desk.
VIAG-1106-E	ERROR LOADING DCR EXIT: EXMP01 = <exitname> The DCR exit identified in the message could not be loaded successfully.	Program Abends	Insure the exit routine identified is in the STEPLIB concatenation of the job step.
VIAG-1107-E	TOO MANY SS-CTRLS FOR THIS TASK. MAX IS <number> More SS-CTRLS were encountered than can be supported. The maximum number allowable is shown in the message.	Program Abends	Report Problem and all relevant information (including trace output, if available) to the ASG Service Desk.

Message	Description and Explanation	System Action	User Action
VIAG-1108-E	DYNAMIC ALLOCATION ERROR ON SCRLIB R15 = <value> REASON = <value> INFO = <value> DDNAME = <ddname> An error was encountered attempting to dynamically allocate the SCR library.	Program Abends	Research the reason and info codes in the message and resolve the problem as required.
VIAG-1109-E	OPEN ERROR FOR SCRLIB R15 = <value> DDNAME = <ddname> DSNAME = <dsname> An error was encountered attempting to open the SCR library.	Program Abends	Research the return code provided and resolve the problem indicated.
VIAG-1110-E	NO MATCHING SCR FOR SS-CTRL ADDRESS <address> Internal Error	Program Abends	Report Problem and all relevant information (including trace output, if available) to the ASG Service Desk.
VIAG-1111-E	CHANGE OF BRIDGE TYPE FROM <value> NOT ALLOWED The program attempted to change the type of bridging during execution.	Program Abends	The program may initiate either forward or reverse bridging, and may not change it subsequently.
VIAG-1112-E	UNSUCCESSFUL CALL TO USE-SCR #INSCRNM = <scrname> SCRU = <name> Internal Error	Program Abends	Report problem and all relevant information (including trace output, if available) to the ASG Service Desk.

Message	Description and Explanation	System Action	User Action
VIAG-1113-E	GETMAIN ERROR FOR NEW I/O AREA. R15 = <value> An error occurred attempting to allocate additional memory.	Program Abends	Increase the memory available to the jobs step. If problem reoccurs, report problem and all relevant information (including trace output, if available) to the ASG Service Desk.
VIAG-1114-E	GETMAIN ERROR FOR NEW KEY AREA. R15 = <value> An error occurred attempting to allocate additional memory.	Program Abends	Increase the memory available to the jobs step. If problem reoccurs, report problem and all relevant information (including trace output, if available) to the ASG Service Desk.
VIAG-1115-E	SS-CTRL ADDRESS <value> NOT FOUND IN SSCTAB. Internal Error	Program Abends	Report problem and all relevant information (including trace output, if available) to the ASG Service Desk.
VIAG-1116-E	SSCREC TABLE HOLDS NO ADDRESS FOR RECORD <recordname>. NO BIND RECORD DONE? Possible program logic error.	Program Abends	Ensure that the program is coded correctly. If the problem reoccurs, report problem and all relevant information (including trace output, if available) to the ASG Service Desk.

Message	Description and Explanation	System Action	User Action
VIAG-1117-E	CONVERT ROUTINE RETURNED NON-ZERO RC = <value> A data conversion routine detected an error.	Program Abends	Ensure that the data is correct, and that the Bridge Definition is correctly defined.
VIAG-1118-E	ERROR FROM GET-REC-INFO. R15 = <value> Internal Error	Program Abends	Report problem and all relevant information (including trace output, if available) to the ASG Service Desk.
VIAG-1119-E	ERROR FROM GET-SET-INFO. R15 = <value> Internal Error	Program Abends	Report problem and all relevant information (including trace output, if available) to the ASG Service Desk.
VIAG-1120-E	CANNOT LOCATE <value1> <value2> OF SUBSCHEMA <subschema name> IN SCR <scrname> Internal Error	Program Abends	Report problem and all relevant information (including trace output, if available) to the ASG Service Desk.
VIAG-1121-E	<value1> of <value2> OF SUBSCHEMA <subschemaname> IN SCR <scrname> NOT FOUND IN DCR <dcrname> Internal Error	Program Abends	Report problem and all relevant information (including trace output, if available) to the ASG Service Desk.

Message	Description and Explanation	System Action	User Action
VIAG-1122-E	CANNOT CONVERT <value1> <value2> … Internal Error	Program Abends	Report problem and all relevant information (including trace output, if available) to the ASG Service Desk.
VIAG-1123-E	CANNOT LOCATE KEY FOR INDEXED SET <value> Internal Error	Program Abends	Report problem and all relevant information (including trace output, if available) to the ASG Service Desk.
VIAG-1124-E	CANNOT LOCATE KEY OF INDEXED SET <value> Internal Error	Program Abends	Report problem and all relevant information (including trace output, if available) to the ASG Service Desk.
VIAG-1125-E	CANNOT LOCATE FIELD <fieldname> FOR INDEXED SET <setname> Internal Error	Program Abends	Report problem and all relevant information (including trace output, if available) to the ASG Service Desk.
VIAG-1126-E	CANNOT LOCATE FIELD <fieldname> OF INDEXED SET <setname> Internal Error	Program Abends	Report problem and all relevant information (including trace output, if available) to the ASG Service Desk.

Appendix C
Common Commands

This table lists the primary commands and where they can be used in Bridge:

Command	Screen/Pop-up
CANCEL	ALL
END	ALL
EXIT	ALL
FIND	Bridge Definition - Source Record View - Bridge Definition Directory View - Bridge Generate Libraries View - Bridge Generate Library Members View - Bridge Definition Cross Reference View/Print - Bridge Event Log
GENERATE	ALL
HELP	ALL
KEYS	ALL
LOCATE	ALL
LPRINT	File - New Bridge Generate Library View - Bridge Generate Library Members View - Bridge Definition Cross Reference
PLIST	ALL
PLOG	ALL
PRINTLIST	ALL
PRINTLOG	ALL

Command	Screen/Pop-up
PRINTLST	ALL
PRODLVL	ALL
RECALL	ALL
RESET	ALL
RETURN	ALL
RFIND	Bridge Definition - Source Record View - Bridge Definition Directory View - Bridge Generate Libraries View - Bridge Generate Library Members View - Bridge Definition Cross Reference View/Print - Bridge Event Log
SAVE	ALL
SELECT	ALL
UNSELECT	ALL
ZIN ZOOMIN ZOOMOUT ZOOM ZOUT	Bridge Definition - Source Record

Appendix D
Default Bridge Parameters and Rules

These tables define the elements of the Bridge Parameter and Bridge Rule Starter Sets as initially distributed with Bridge.

Bridge Parameter Starter Set

These are the elements of the Bridge Parameter Starter Set:

Name	Macro Keyword	Description	Usage	Value Type	Value Length	Default Value	Allowed Value
FILL	FILCHR	Fill character	VALUE	KEYW	1	SPACES	SPACES LOW VALUES HIGH VALUES ZEROS
ISCLEAN	CLEAN	Data Field is "Clean"	VALUE	CHAR	N/A	NO	YES, NO
RATE	EURO	Currency Conversion Rate	VALUE	NUMB	7	N/A	N/A
RATEO	EXHCO	Triangulation Output Rate	VALUE	NUMB	7	N/A	N/A
RNDLOSS	RL	Rounding Loss (NC-to-Euro Result)	OUTPUT	N/A	N/A	N/A	N/A
RNDLOSSI	RL	Rounding Loss (Euro -to-NC Input)	INPUT	N/A	N/A	N/A	N/A
USEURO	EXCH	US to Euro Conversion Rate	VALUE	NUMB	7	0001000	N/A
WINYEAR	WIN	Window Year	VALUE	NUMB	2	50	N/A

Bridge Rule Starter Set

These are the parameters of the Bridge Rule Starter Set:

Name	Description	Data Type	Old Length	New Length	Macro Name	Parameters
ADJUST	Change the length of a (Filler) item.	ALPNM	5	7	DDDYY	ISCLEAN WINYEAR
DDDYYB	Convert 'DDDYY' Dates to 'DDDYYYY' (Binary Fullword)	BNRY	4	4	DDDYYB	ISCLEAN WINYEAR
DDDYYP	Convert 'DDDYY' Dates to 'DDDYYYY' (Packed Decimal)	PACK	3	4	DDDYYP	ISCLEAN WINYEAR
DDMMYY	Convert 'DDMMYY' Dates to 'DDMMYYYY' (Zoned Dec.)	NUM	6	8	DDMMYY	ISCLEAN WINYEAR
DDMMYYB	CONVERT 'DDMMYY' DATES TO 'DDMMYYYY' (BINARY FULLWORD)	BNRY	4	4	DDMMYYB	ISCLEAN WINYEAR
DDMMYYP	CONVERT 'DDMMYY' DATES TO 'DDMMYYYY' (PACKED DECIMAL)	PACK	4	5	DDMMYYP	ISCLEAN WINYEAR
EURO2USB	CONVERT EUROS TO US DOLLARS (BINARY FULLWORD)	BNRY	4	4	EURO2NCB	RNDLOSSI USEURO
EURO2USP	CONVERT EUROS TO US DOLLARS (PACKED DEC.)	PACK	ANY	ANY	EURO2NCP	RNDLOSSI USEURO
EURO2USZ	CONVERT EUROS TO US DOLLARS (ZONED DEC.)	NUM	ANY	ANY	EURO2NCZ	RNDLOSSI USEURO
FDDMMYY	CONVERT 'DD/MM/YY'TO 'DD/MM/YYYY' (ZONED DEC.)	NUM	8	10	FDDMMYY	ISCLEAN WINYEAR
FMMDDYY	CONVERT 'MM/DD/YY' DATES TO 'MM/DD/YYYY' (ZONED DEC.)	NUM	8	10	FDDMMYY	ISCLEAN WINYEAR
FYYMMDD	CONVERT 'YY/MM/DD' DATES TO 'YYYY/MM/DD' (ZONED DEC.)	NUM	8	10	FYYMMDD	ISCLEAN WINYEAR
MDY2YMD	CONVERT 'MMDDYY' DATES TO 'YYYYMMDD' (ZONED DEC.)	NUM	6	8	MDY2YMD	ISCLEAN WINYEAR

Name	Description	Data Type	Old Length	New Length	Macro Name	Parameters
MMDDYY	CONVERT 'MMDDYY' DATES TO 'MMDDYYYY' (ZONED DEC.)	NUM	6	8	DDMMYY	ISCLEAN WINYEAR
MMDDYYB	CONVERT 'MMDDYY' DATES TO 'MMDDYYYY' (BINARY FULLWORD)	BNRY	4	4	MMDDYYB	ISCLEAN WINYEAR
MMDDYYP	CONVERT 'MMDDYY' DATES TO 'MMDDYYYY' (PACKED DECIMAL)	PACK	4	5	DDMMYYP	ISCLEAN WINYEAR
MMYY	CONVERT 'MMYY' DATES TO 'MMYYYY' (ZONED DEC.)	NUM	4	6	DDDYY	ISCLEAN WINYEAR
MMYYBF	CONVERT 'MMYY DATES TO 'MMYYYY (BINARY FULLWORD)	BNRY	4	4	MMYYB	ISCLEAN WINYEAR
MMYYBH	CONVERT 'MMYY' DATES TO 'MMYYYY' (BINARY HALFWORD)	BNRY	2	4	MMYYB	ISCLEAN WINYEAR
MMYYP	CONVERT 'MMYY' DATES TO 'MMYYYY' (PACKED DECIMAL)	PACK	3	4	DDMMYYP	ISCLEAN WINYEAR
NCA2NCBZ	NC 'A' TO NC 'B' - TRIANGULATION DEMONSTRATION	NUM	ANY	ANY	NC2NCZ	IS CLEAN RATEI RATEO
US2EUROB	CONVERT US DOLLARS TO EURO (BINARY FULLWORD)	BNRY	4	4	NC2EUROB	RNDLOSS USEURO
US2EUROP	CONVERT US DOLLARS TO EURO (PACKED DEC.)	PACK	ANY	ANY	NC2EUROP	RNDLOSS USEURO
US2EUROZ	CONVERT US DOLLARS TO EURO (ZONED DEC.)	NUM	ANY	ANY	NC2EUROZ	RNDLOSS USEURO
YY	CONVERT 'YY DATES TO 'YYYY' (ZONED DEC.)	NUM	2	4	YY	ISCLEAN WINYEAR
YYB	CONVERT 'YY' DATES TO 'YYYY' (BINARY HALFWORD)	BNRY	2	2	YYB	ISCLEAN WINYEAR
YYDD	CONVERT 'YYDDD' DATES TO 'YYYYDDD' (ZONED DEC.)	NUM	5	7	YYDDD	ISCLEAN WINYEAR

Name	Description	Data Type	Old Length	New Length	Macro Name	Parameters
YYDDDB	CONVERT 'YYDDD' DATES TO 'YYYYDDD' (BINARY FULLWORD)	BNRY	4	4	YYDDDB	ISCLEAN WINYEAR
YYDDDP	CONVERT 'YYDDD' DATES TO 'YYYYDDD' (PACKED DECIMAL)	PACK	3	4	YYDDDP	ISCLEAN WINYEAR
YYMM	CONVERT 'YYMM' DATES TO 'YYYYMM' (ZONED DEC.)	NUM	4	6	YYDDD	ISCLEAN WINYEAR
YYMMBF	CONVERT 'YYMM' DATES TO 'YYYYMM' (BINARY FULLWORD)	BNRY	4	4	YYMMB	ISCLEAN WINYEAR
YYMMBH	CONVERT 'YYMM' DATES TO 'YYYYMM' (BINARY HALFWORD)	BNRY	2	4	YYMMB	ISCLEAN WINYEAR
YYMMDD	CONVERT 'YYMMDD' DATES TO 'YYYYMMDD' (ZONED DEC.)	NUM	6	8	YYMMDD	ISCLEAN WINYEAR
YYMMDDB	CONVERT 'YYMMDD' DATES TO 'YYYYMMDD' (BINARY FULLWORD)	BNRY	4	4	YYMMDDB	ISCLEAN WINYEAR
YYMMDDP	CONVERT 'YYMMDD' DATES TO 'YYYYMMDD' (PACKED DECIMAL)	PACK	4	5	YYMMDDP	ISCLEAN WINYEAR
YYMMP	CONVERT 'YYMM' DATES TO 'YYYYMM' (PACKED DECIMAL)	PACK	3	4	YYMMDDP	ISCLEAN WINYEAR

Appendix E
Bridge Date Conversion Option

These tables define the elements of the Bridge Parameter and Bridge Rule Starter Sets distributed with the date conversion option.

Bridge Parameter Starter Set

These are the elements of the Bridge Parameter Starter Set:

Name	Macro Keyword	Description	Usage	Value Type	Value Length	Default Value	Allowed Value
WINYEAR	WIN	Window Year	VALUE	NUMB	2	50	N/A

Bridge Rule Starter Set

These are the elements of the Bridge Rule Starter Set:

Name	Description	Data Type	Old Length	New Length	Macro Name	Parameters
ADJUST	Change the length of a (Filler) item	ALPNM	5	7	DDDYY	ISCLEAN WINYEAR
DDDYYB	Convert 'DDDYY' Dates to 'DDDYYYY' (Binary Fullword)	BNRY	4	4	DDDYYB	ISCLEAN WINYEAR
DDDYYP	Convert 'DDDYY' Dates to 'DDDYYYY' (Packed Decimal)	PACK	3	4	DDDYYP	ISCLEAN WINYEAR
DDMMYY	Convert 'DDMMYY' Dates to 'DDMMYYYY' (Zoned Dec.)	NUM	6	8	DDMMYY	ISCLEAN WINYEAR

Name	Description	Data Type	Old Length	New Length	Macro Name	Parameters
DDMMYYB	CONVERT 'DDMMYY' DATES TO 'DDMMYYYY' (BINARY FULLWORD)	BNRY	4	4	DDMMYYB	ISCLEAN WINYEAR
DDMMYYP	CONVERT 'DDMMYY' DATES TO 'DDMMYYYY' (PACKED DECIMAL)	PACK	4	5	DDMMYYP	ISCLEAN WINYEAR
FDDMMYY	CONVERT 'DD/MM/YY'TO 'DD/MM/YYYY' (ZONED DEC.)	NUM	8	10	FDDMMYY	ISCLEAN WINYEAR
FMMDDYY	CONVERT 'MM/DD/YY' DATES TO 'MM/DD/YYYY' (ZONED DEC.)	NUM	8	10	FDDMMYY	ISCLEAN WINYEAR
FYYMMDD	CONVERT 'YY/MM/DD' DATES TO 'YYYY/MM/DD' (ZONED DEC.)	NUM	8	10	FYYMMDD	ISCLEAN WINYEAR
MDY2YMD	CONVERT 'MMDDYY' DATES TO 'YYYYMMDD' (ZONED DEC.)	NUM	6	8	MDY2YMD	ISCLEAN WINYEAR
MMDDYY	CONVERT 'MMDDYY' DATES TO 'MMDDYYYY' (ZONED DEC.)	NUM	6	8	DDMMYY	ISCLEAN WINYEAR
MMDDYYB	CONVERT 'MMDDYY' DATES TO 'MMDDYYYY' (BINARY FULLWORD)	BNRY	4	4	MMDDYYB	ISCLEAN WINYEAR
MMDDYYP	CONVERT 'MMDDYY' DATES TO 'MMDDYYYY' (PACKED DECIMAL)	PACK	4	5	DDMMYYP	ISCLEAN WINYEAR
MMYY	CONVERT 'MMYY' DATES TO 'MMYYYY' (ZONED DEC.)	NUM	4	6	DDDYY	ISCLEAN WINYEAR
MMYYBF	CONVERT 'MMYY DATES TO 'MMYYYY (BINARY FULLWORD)	BNRY	4	4	MMYYB	ISCLEAN WINYEAR
MMYYBH	CONVERT 'MMYY' DATES TO 'MMYYYY' (BINARY HALFWOR D)	BNRY	2	4	MMYYB	ISCLEAN WINYEAR

Name	Description	Data Type	Old Length	New Length	Macro Name	Parameters
MMYYP	CONVERT 'MMYY' DATES TO 'MMYYYY' (PACKED DECIMAL)	PACK	3	4	DDMMYYP	ISCLEAN WINYEAR
YY	CONVERT 'YY DATES TO 'YYYY' (ZONED DEC.)	NUM	2	4	YY	ISCLEAN WINYEAR
YYB	CONVERT 'YY' DATES TO 'YYYY' (BINARY HALFWORD)	BNRY	2	2	YYB	ISCLEAN WINYEAR
YYDD	CONVERT 'YYDDD' DATES TO 'YYYYDDD' (ZONED DEC.)	NUM	5	7	YYDDD	ISCLEAN WINYEAR
YYDDDB	CONVERT 'YYDDD' DATES TO 'YYYYDDD' (BINARY FULLWORD)	BNRY	4	4	YYDDDB	ISCLEAN WINYEAR
YYDDDP	CONVERT 'YYDDD' DATES TO 'YYYYDDD' (PACKED DECIMAL)	PACK	3	4	YYDDDP	ISCLEAN WINYEAR
YYMM	CONVERT 'YYMM' DATES TO 'YYYYMM' (ZONED DEC.)	NUM	4	6	YYDDD	ISCLEAN WINYEAR
YYMMBF	CONVERT 'YYMM' DATES TO 'YYYYMM' (BINARY FULLWORD)	BNRY	4	4	YYMMB	ISCLEAN WINYEAR
YYMMBH	CONVERT 'YYMM' DATES TO 'YYYYMM' (BINARY HALFWOR D)	BNRY	2	4	YYMMB	ISCLEAN WINYEAR
YYMMDD	CONVERT 'YYMMDD' DATES TO 'YYYYMMDD' (ZONED DEC.)	NUM	6	8	YYMMDD	ISCLEAN WINYEAR
YYMMDDB	CONVERT 'YYMMDD' DATES TO 'YYYYMMDD' (BINARY FULLWORD)	BNRY	4	4	YYMMDDB	ISCLEAN WINYEAR

Name	Description	Data Type	Old Length	New Length	Macro Name	Parameters
YYMMDDP	CONVERT 'YYMMDD' DATES TO 'YYYYMMDD' (PACKED DECIMAL)	PACK	4	5	YYMMDDP	ISCLEAN WINYEAR
YYMMP	CONVERT 'YYMM' DATES TO 'YYYYMM' (PACKED DECIMAL)	PACK	3	4	YYMMDDP	ISCLEAN WINYEAR

Appendix F
Bridge for Euro Option

The Bridge for Euro option provides data conversion utilities for euro implementation. It enables you to translate currency fields in data, screens and reports between national currencies (through triangulation), as well as between euro and national currencies.

Note:
If you are using the Bridge for Euro option, you need to review the information in this section.

Bridge Parameter Starter Set

These tables define the elements of the Bridge Parameter Starter Set distributed with the Bridge for Euro option. Use these parameters for fixed exchange rate macros:

Name	Macro Keyword	Description	Usage	Value Type	Value Length	Default Value	Allowed Value
RATE	EURO	Currency Conversion Rate	VALUE	NUMB	7	N/A	N/A
RATEO	EXHCO	Triangulation Output Rate	VALUE	NUMB	7	N/A	N/A
RNDLOSS	RL	Rounding Loss (NC-to-Euro Result)	OUTPUT	N/A	N/A	N/A	N/A
RNDLOSSI	RL	Rounding Loss (Euro-to-NC Input)	INPUT	N/A	N/A	N/A	N/A
USEURO	EXCH	US to Euro Conversion Rate	VALUE	NUMB	7	0001000	N/A

Use these parameters for table-driven exchange rate macros:

Name	Macro	Required	Description
ESICUR*	NCTEURO NCTNC	No	Source Currency Name in Input Field. You must use ESVCUR or ESICUR.
ETICUR*	NCTEURO NCTNC	No	Target Currency Name in Input Field. You must use ETVCUR or ETICUR.
ESVCUR*	NCTEURO	No	Source Currency Name as Value. If you omit this value, ESICUR is used.
ETVCUR	NCTNC	No	Target Currency Name as Value
ESVDEC*	EUROTNC NCTNC	No	Source Currency Scaling Factor (# of decimal places). If you omit this value, the currency table value is used.
ETVDEC*	EUROTNC NCTNC	No	Target Currency Scaling Factor (# of decimal places). If you omit this value, the currency table value is used.
EIVDEC*	NCTNC	No	Intermediate Result Scaling Factor (for triangulation). The default value is 2.
EDOSRC	NCTEURO NCTNC	No	Rounding Loss in Source Currency Field. This field is optional. However, if you want to use the rounding loss, you must assign to this parameter the field that receives the value.
EDITGT	EUROTNC	No	Rounding Loss in Target Currency Field. This is an input field.
EIOAMT		No	Intermediate Result Output field (for triangulation).
FORMAT	All	No	Indicates data format. The valid values are Z (zoned decimal), P (packed decimal), or B (binary). The default is Z.
CHECK	All	No	Check data for valid values. The default is No.
CLEAN	All	No	Checks for initialized data; blanks, high-values, low-values. The default is No.

Note:

The macro names indicate the type of currency conversion: NCTNC = national currency to national currency (triangulation); EUROTNC = euro to national currency; and NCTEURO = national currency to euro.

Bridge Rule Starter Set

This table defines the elements of the Bridge Rule Starter Set distributed with the Bridge for Euro option. You can copy and customize these rules for specific currencies:

Name	Description	Data Type	Old Length	New Length	Macro Name
ACTEUROZ	Any currency to euro	Zoned	Any	Any	NCTEURO
NCTEUROZ	National currency to euro	Zoned	Any	Any	NCTEURO
NCTNCZ	National currency to national currency	Zoned	Any	Any	NCTNC
EUROTNCZ	Euro to national currency	Zoned	Any	Any	EUROTNC

When defining a Bridge rule for currency conversion, you can select the scaling factor parameter or allow Bridge to obtain it from the currency table. Enabling Bridge to obtain this data is useful if your currency table contains multiple currencies with different scaling factors. If you choose to pass this parameter, the scaling factor in the currency table is ignored. Use the parameters ESVDEC (source currency scaling factor), or ETVDEC (target currency scaling factor) when defining your Bridge rule.

Index

Z